



AD 760692

RANDI: RESEARCH AMBIENT NOISE DIRECTIONALITY MODEL

by R. A. Wagstaff

Undersea Surveillance and Ocean Sciences Department

April 1973



Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
U.S. Department of Commerce
Springfield, VA 22151

DDC
RECEIVED
MAY 30 1973
B

Approved for public release; distribution unlimited.

83
R



NAVAL UNDERSEA CENTER, SAN DIEGO, CA. 92132

AN ACTIVITY OF THE NAVAL MATERIAL COMMAND

ROBERT H. GAUTIER, CAPT, USN

Commander

Wm. B. McLEAN, Ph.D.

Technical Director

ADMINISTRATIVE STATEMENT

The work reported herein was performed under NAVSHIPS Project No. SF 55 2070 by members of the Acoustic Environmental Modeling Division. The paper covers work from June 1971 to October 1972.

Released by
L. K. ARNDT, Head
Acoustic Environmental
Modeling Division

Under authority of
G. B. ANDERSON, Head
Undersea Surveillance and
Ocean Sciences Department

ACCESSION for		
NTIS	White Section	<input checked="" type="checkbox"/>
BBC	Buff Section	<input type="checkbox"/>
UNANNOUNCED		<input type="checkbox"/>
JUSTIFICATION		
BY		
DISTRIBUTION/AVAILABILITY CODES		
Dist.	AVAIL. and/or SPECIAL	
A		

SUMMARY

PROBLEM

Develop a computer model for calculating the vertical and horizontal directionality of low-frequency ambient noise for an ocean environment.

RESULTS

A FORTRAN computer model has been developed. The Research Ambient Noise Directionality Model (RANDI) has given results in good agreement with measured data for the Pacific, Atlantic, and the Mediterranean.

RECOMMENDATIONS

RANDI noise calculations should be compared with ambient noise measurements for other ocean areas, seasons, and noise-source distributions as data, sufficiently documented for validation purposes, become available.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Naval Undersea Center San Diego, Calif. 92132		2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	
		2b. GROUP	
3. REPORT TITLE RANDI: RESEARCH AMBIENT NOISE DIRECTIONALITY MODEL			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Research report, June 1971-October 1972			
5. AUTHOR(S) (First name, middle initial, last name) R. A. Wagstaff			
6. REPORT DATE April 1973		7a. TOTAL NO. OF PAGES 82	7b. NO. OF REFS 13
8a. CONTRACT OR GRANT NO.		9a. ORIGINATOR'S REPORT NUMBER(S) NUC TP 349	
b. PROJECT NO. NAVSHIPS Project No. SF 55 2070			
c.		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d.			
10. DISTRIBUTION STATEMENT Approved for public release; distribution unlimited.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY Naval Ship Systems Command Washington, D.C. 20360	
13. ABSTRACT A computer model for calculating the vertical and horizontal directionality of ambient noise and some of the results which have been obtained are discussed. The model considers three sources of surface-generated anisotropic noise, one source of surface-generated isotropic noise and two sources of volumetric isotropic noise. Noise levels are obtained for various propagation conditions and for different sensor depths. Results agree well with experimental data, and it is shown that the model can be a useful tool in the planning of ambient noise measurement experiments and in the analysis of results.			

UNCLASSIFIED

Security Classification

iia

UNCLASSIFIED

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Ambient noise Vertical directionality Horizontal directionality Noise calculation Computer model Depth dependence Noise sources Target noise Underwater acoustics						

11b

CONTENTS

INTRODUCTION	1
BACKGROUND	1
AMBIENT NOISE DIRECTIONALITY MODEL	3
Model Description	3
Mathematical Description	4
Computational Procedure	8
Model Calibration Procedure	9
Model Validation	9
DISCUSSION OF RESULTS	10
General Results	10
Model Output–Measured Data Comparison	14
FUTURE IMPROVEMENTS	16
SUMMARY	16
REFERENCES	18
Appendixes:	
A. RANDI SUBROUTINE DESCRIPTIONS AND FLOW DIAGRAM	19
B. RANDI INPUT–OUTPUT	22
C. RESEARCH AMBIENT NOISE DIRECTIONALITY (RANDI) MODEL LISTING	40

Preceding page blank

INTRODUCTION

The Research Ambient Noise Directionality Model (RANDI) is an operating general-purpose FORTRAN Program that calculates and displays, via CALCOMP plots, the vertical and horizontal directionality of low-frequency (between 10 and 500 Hz) ambient noise for an ocean environment. This first-phase program will be used in the design of ambient noise measurement experiments and in surveillance systems analysis studies. A variable-dependent approach based on system and environmental parameter variance makes this model inappropriate for synoptic forecasting. It will be improved and validated as time, data, and state-of-the-art permit.

BACKGROUND

The ambient noise at any particular point in the ocean depends upon the amount and location of noise-generating sources and the nature of the local acoustic propagation conditions, i.e., how the noise gets from the generator to the measuring sensor.

Sound in the ocean travels in curved or refracted paths and will arrive at a hydrophone from various vertical angles, depending on the depth of the source and the hydrophone and on the separation range. The path an individual sound ray travels can include surface reflections and bottom bounces, as illustrated in the ray diagram (Fig. 1). The bottom bounce paths will generally have the highest sound energy losses, while the paths which do not make contact with either the surface or the bottom will have the smallest transmission losses. The vertical angle between one of these latter rays and the horizontal is generally less than 15 deg at any range, while the rays making contact with the surface or bottom can have vertical angles approaching 90 deg. This effectively divides the sound rays into three distinct groups: (1) rays which come in contact with the bottom and arrive at angles from approximately 15 to 90 deg down from the horizontal; (2) the near-horizontal or SOFAR Channel rays, which touch neither surface nor bottom and, (3) the rays arriving at angles above 15 deg and which originated or were reflected from near the surface.

There are many sources of noise, the most important being winds, waves, shipping, thermal agitation and biological and seismic activity. The noise due to winds, waves, and shipping originates near the surface and arrives at the sensor along paths which travel near or reflect from the surface. In the case of distant sources, the noise travels by way of the SOFAR Channel and will arrive, at a sensor located within the channel, from nearly horizontal angles. Noise from near sources (within a few hundred miles) will arrive at angles closer to the vertical than does the SOFAR Channel noise. Thermal noise comes from throughout the medium and can arrive at any angle or from any direction. Seismic noise would be expected to arrive along those paths which come in contact with radiating or reflecting surfaces, including the bottom and distant seamounts. Biological noise, however, originates along the surface, the bottom, and throughout the medium.

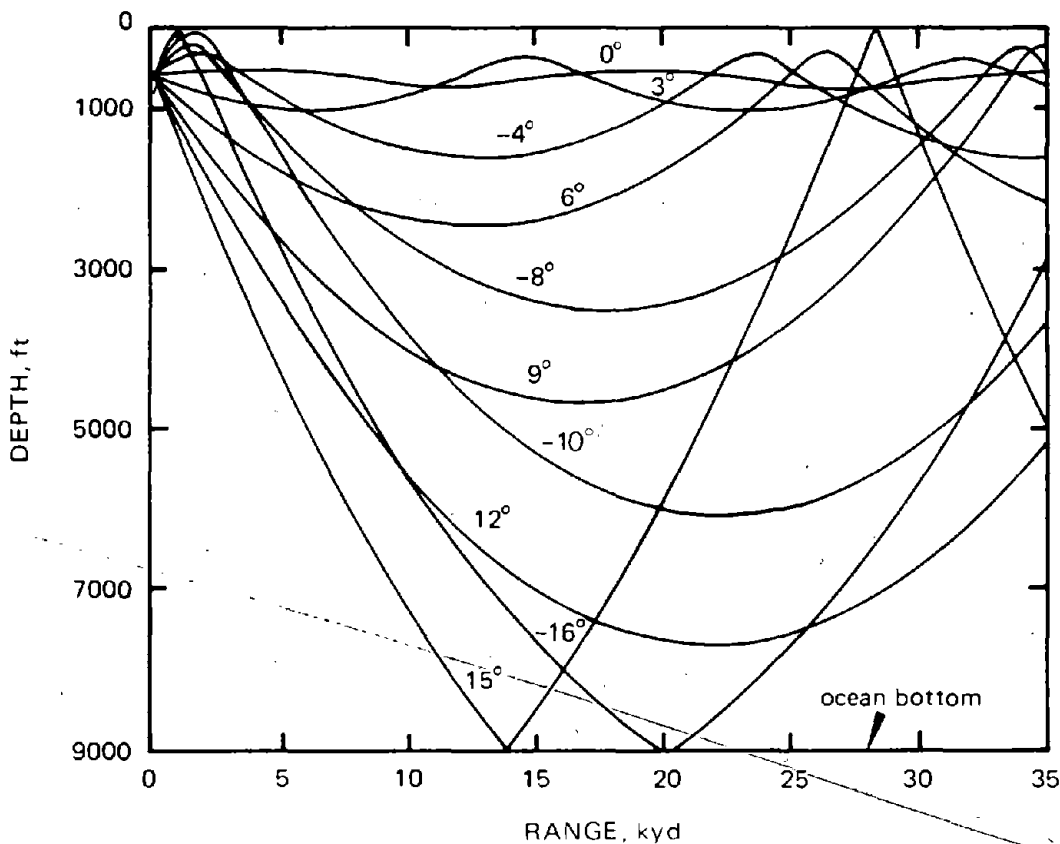


Figure 1. Typical raytrace diagram.

The net effect of propagating—to the sensor—the noises from such a highly stratified distribution of noise generators is a noise field having vertical directionality, i.e., noise level dependent on vertical arrival angle, as illustrated in Fig. 2.

Various ambient noise models have been proposed which include part of the major noise sources and invoke certain simplifying assumptions to produce a medium less hostile to mathematical description in an attempt to render the problem solvable and yet obtain meaningful results. Few investigators have succeeded, and then only in highly specialized cases. There is documentation on several of these ambient noise directionality models. In the opinion of the author, none is adequate for use in the design of detailed directional noise measurement experiments or for use in surveillance systems analysis studies. Miller (Ref. 1) and Urick (Ref. 2), for example, proposed theoretical models which require a zero sound-speed gradient. These models cannot account for surface-generated noise arriving at angles below the horizontal or for bottom noise (biological, seismic) arriving at angles above the horizontal, except by reflection from the bottom or surface, respectively. This results in a distorted directional noise pattern with abnormally low levels near the horizontal. Talham's model (Ref. 3), on the other hand, includes realistic sound-speed profiles but applies only to bottom-mounted hydrophones. Bartberger (Ref. 4) considered only ship

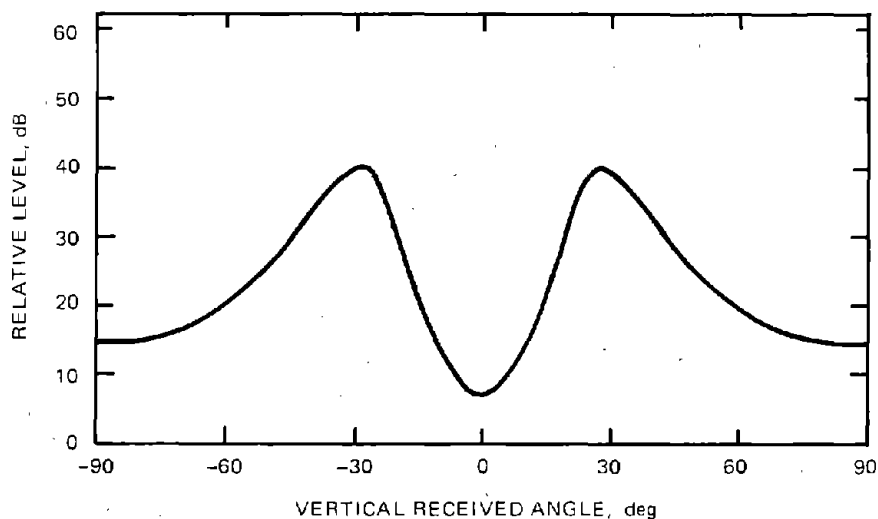


Figure 2. Typical ambient noise vertical directionality plot.

noise from a uniform distribution of surface ships, thus restricting the usefulness of his model to the 20- to 100-Hz region. Finally, the MOSS Ambient Noise Model developed jointly by the Navy and Bell Telephone Laboratories calculates the expected vertical directionality of the noise field for seven ocean areas for the mean summer and winter conditions for acoustic propagation, wind speed, and shipping. A more general area-independent model is desired.

In addition to the limitations already discussed, none of the above models considers horizontal directionality, noise due to distant sources (SOFAR Channel noise), or noise of a biological or transient nature, nor does any contain sufficient provision for considering variances in system characteristics. For these reasons RANDI was developed for use in the design of ambient noise measurement experiments and the analysis of surveillance systems.

AMBIENT NOISE DIRECTIONALITY MODEL

MODEL DESCRIPTION

RANDI utilizes one or two of three different sources for the propagation loss between the noise generators and the sensor. The first source is a self-contained linear raytrace routine, one which approximates the sound-speed profile by a series of straight-line segments and corrects for earth curvature. See Ref. 5 for a detailed description.

The second source is a set of propagation loss versus range and arrival angle arrays which is input as data. If the propagation loss is input, RANDI will bypass the raytrace routine. Hence, ambient noise calculations by RANDI can be based on propagation loss values from ray theory (its own or nearly any other raytrace model, including those that account for variable bottom topography and horizontal changes in the sound-speed profile),

normal-mode theory, experimental measurements, or any method by which propagation loss versus range and arrival angle might be obtained. This second source of propagation loss can be extremely useful when operating RANDI at the relatively low frequencies of many passive surveillance systems, where the validity of ray theory becomes questionable and normal-mode theory more attractive.

The third form of propagation loss utilized by RANDI is for sound energy traveling from distant sources to a sensor located within the SOFAR Channel. RANDI calculates the propagation loss in this case by considering the effects of frequency-dependent attenuation and spreading loss increasing with fifteen times the log of the average range to the continental shelf or the range to where the SOFAR Channel reaches the surface (as in northern latitudes).

In addition to distant noise, RANDI considers five other sources of isotropic and anisotropic surface and volumetric noise (Fig. 3): shipping, sea state 0, biological, rain, and wind-wave interaction.

The surface noise is generated by an infinite number of point sources distributed along a horizontal "noise source plane" just below the surface (Fig. 4). The depth of this plane is set at 20 ft, since the main ship noise radiators (screws, shaft, and hull) are near this depth, and surface wave action extends well below the surface. The shipping noise generators are nonuniformly distributed, while the wind-wave and rain noise generators are uniformly distributed. The noise resulting from any of these three sources is anisotropic. Sea state 0 and biological noises come from a uniform volumetric distribution of noise generators centered around the sensor and result in isotropic noise. The SOFAR Channel noise received by a sensor located within the SOFAR Channel is also isotropic for all channeled ray angles. The total noise field, then, is part isotropic and part anisotropic.

A target capability is also included in RANDI. By specifying the oceanographic, environmental and noise conditions, and a target location, depth, and frequency spectrum or line component, the Surveillance System Analyst receives, by way of plot and printed output, the levels and angles of target multiple arrivals superimposed on the ambient noise arrivals (see Figs. B-4 through B-13). Such a capability can make RANDI a useful tool in the design, analysis, and optimization of surveillance systems.

MATHEMATICAL DESCRIPTION

The anisotropic shipping noise squared pressure spectrum level SNL (for a 1-Hz band) in the model is obtained from surface noise generators which have a radiated sound pressure level varying with frequency. The following empirical expression, which has characteristics similar to the spectra given in Fig. 71 of Ref. 6, is utilized:

$$\begin{aligned} \text{SNL (re } \mu\text{Pa)} = & A_0 - 10 \log (10^{-1.0 \log f' + 1.16} + 10^{+3.3 \log f' - 6.27}) \\ & + 4 \text{ SHIPD} + 50 \log (\text{SPEED}/12) + 20 \log (\text{LENGTH}/300) \end{aligned}$$

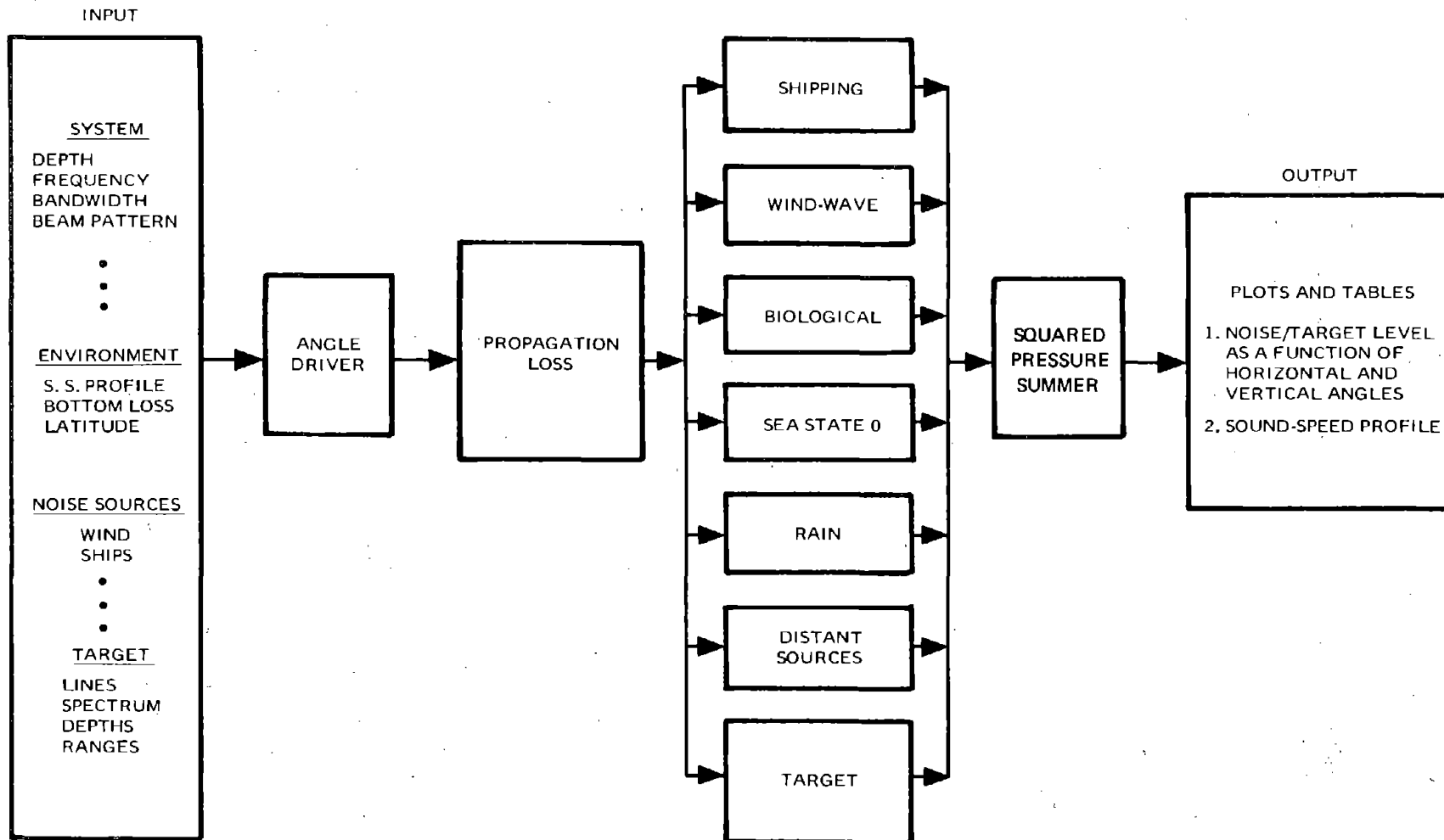


Figure 3. RANDI model block diagram.

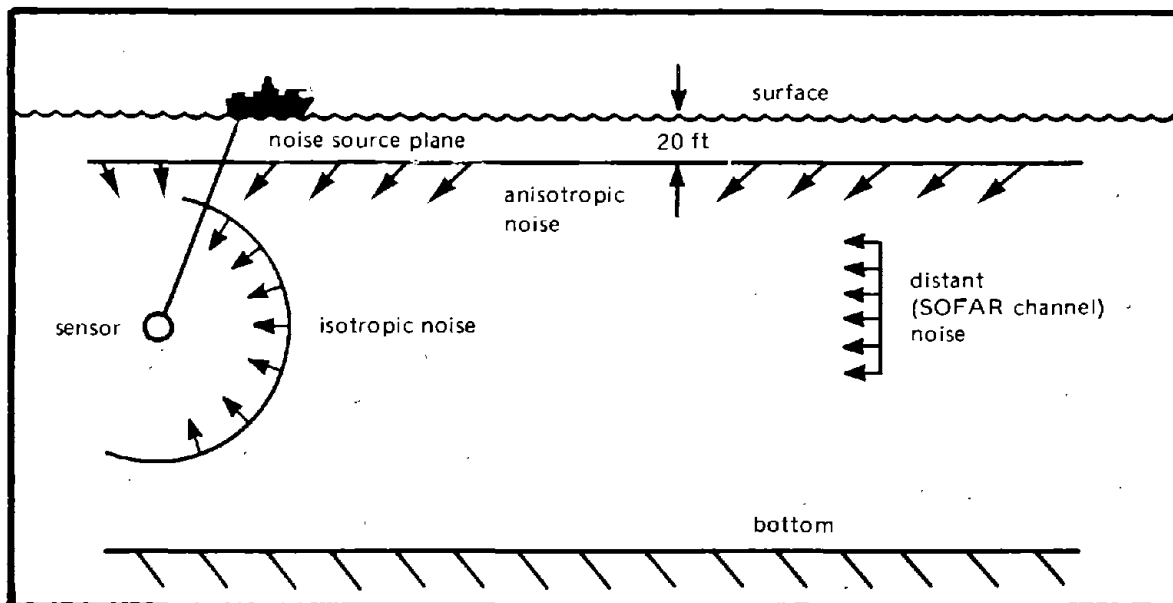


Figure 4. RANDI model noise field.

where

$$A_0 = \begin{cases} 90 & \text{no distant noise sources} \\ 85 & \text{distant noise sources specified} \end{cases}$$

$$f' = f - 2 \text{ SHIPD} + 12$$

f = number of hertz

SHIPD = shipping density indicator (0 – 7)

SPEED = number of knots of noise-source ships' average speed

LENGTH = number of feet of noise-source ships' average length

The squared pressures obtained from the levels given by this function are distributed over the noise source plane in such a way to result in effective squared pressures for unit area at unit distance (the mechanics are described in the section on model calibration) with magnitude varying in range and azimuth in such a manner to be proportional to the density of an input shipping density distribution ASHIP (ship density vs range array) or a shipping density indicator SHIPD (no shipping to heavy shipping on a scale from zero to seven). A similar expression was used for the squared pressure level due to distant sources DSL, since shipping is the major contributor at low frequencies and the higher frequencies are severely attenuated at SOFAR Channel propagation ranges.

The squared pressure spectrum levels for the anisotropic noise WNL due to wind-wave interaction were obtained from a fit to the data of Perrone (Ref. 7) and are a function of wind speed and frequency as follows:

$$\begin{aligned} \text{WNL (re } \mu\text{Pa)} &= -18 \log f + 96 \\ &+ [9.66 (\log f)^2 - 26 \log f + 27.3] (0.065 \text{ WSPD}^{0.7}) \end{aligned}$$

where WSPD = number of knots of wind speed in the vicinity of the sensor.

These levels are converted into mean squared sound pressure in a 1-Hz band and further modified by a multiplying factor to achieve directionality of the source at high frequencies and nondirectionality at the lower frequencies as indicated by Becken (Ref. 8). The wind noise squared pressure modifier used in the model is the following:

$$\text{Squared pressure multiplier} = 1 - (1 - \cos^n \phi) (0.002 f - 0.02)$$

where ϕ = angle (in degrees) from the vertical that the ray makes at the source, and

$$n = \begin{cases} \frac{90 - \phi}{10} & 0 \leq \phi \leq 70 \\ 2 & 70 \leq \phi \leq 90 \end{cases}$$

The following function, obtained from the data of Franz (Ref. 9), gives, for rain, the expected noise squared pressure level RPL in a 1-Hz band and is distributed uniformly along the horizontal noise plane in a manner similar to the wind-wave noise.

$$\text{RPL (re } \mu\text{Pa)} = 5.5 \log f + 50.5 + 14.5 \log (\text{RAIN})$$

where RAIN is the number of inches of rainfall per hour in the vicinity of the sensor.

Isotropic sea state 0 noise squared pressure spectrum level SS0 is that noise which is measured under the ideal conditions of no wind, calm surface, no biological activity, and negligible shipping. It varies with frequency and is independent of depth and geographic location. The equation for SS0 (for a 1-Hz band) was obtained by a quadratic fit to data of Wenz (Ref. 10):

$$\text{SS0 (re } \mu\text{Pa)} = 4.22 (\log f)^2 - 33.4 \log f + 89.1$$

A volumetric or isotropic distribution was chosen for biological noise in the absence of data indicating otherwise. This noise varies with time of day and relative amount of activity expected at a particular site. An activity indicator on a scale from zero to ten is an input to the model. The equation for the biological noise squared pressure spectrum level BPL is of the following form:

$$\begin{aligned} \text{BPL (re } \mu\text{Pa)} &= 0.00175 (100 - f) \sin [0.00262 (\text{hr} - 300)] \text{ ACTIVITY} \\ &- 16.96 (\log f)^2 + 50.1 \log f + 45.1 \end{aligned}$$

where the $0 \leq \text{ACTIVITY} \leq 10$ and hr is the local time of day (military designation 0000–2400).

The sound pressure spectrum level TSL for a target source is accepted by the model by reading in the appropriate coefficients A_i , $i = 1, 3$, of a third-degree polynomial

$$\text{TSL} = A_3 (\log f)^3 + A_2 (\log f)^2 + A_1 \log f + A_0$$

In the event that the line component is the dominant feature in the band of interest, A_0 is the level of the line. Also specified by the user are the initial target range and depth, the numbers of target ranges and depths to be calculated, and the increments in range and depth to be used.

COMPUTATIONAL PROCEDURE

All of the noise and target squared pressure spectrum levels are integrated over a user-specified bandwidth by means of an input frequency response function. If none is specified, a 1-Hz bandwidth and a constant bandpass frequency response function are assumed.

The squared pressure received for a differential vertical angle is obtained by first calculating the area defined by the intersections of the corresponding ray bundle with the noise source plane. These areas are multiplied by the local effective squared noise pressures for unit area at a distance of 1 yard. The resulting squared pressures are then reduced to account for propagation loss and summed. To this value is added the contribution of the isotropic noise sources. The result is further reduced to account for the vertical response of an individual hydrophone or an array of hydrophones. If no response function is specified, it is assumed omnidirectional. The final squared noise pressure arriving at that angle is stored for output and the process is repeated for an adjacent ray bundle at a new vertical angle.

In the case where horizontal directionality is desired, the ocean is divided into n regions by passing vertical planes through the sensor location at $360/n$ deg. The ocean is then effectively divided into n regions, similar to a huge sliced pie of infinite radius whose thickness is equal to the ocean depth at the receiver location. The pertinent environmental and noise parameters are specified separately for each "pie slice" region. Hence, the calculations performed for one region are independent of the calculations performed in adjacent regions. The total squared noise pressure, that which would be measured by an omnidirectional hydrophone, is obtained by summing the squared noise pressures for the n independent "pie slice" regions. An example is given in Appendix B, Figs. B-6 through B-11.

An explanation of terminology used in the ambient noise directionality illustrations is necessary before results can be interpreted. Noise level refers to the per steradian mean squared pressure spectrum level. Vertical received angle is the angle at the sensor the incident ray makes with the horizontal. The negative rays are downcoming rays at the sensor, with the ray having a vertical received angle of -90 deg being the ray which arrives from

the surface directly above the sensor. The rays with positive angles arrive at the sensor from angles below the horizontal.

Unlike noise, targets are treated as point sources at specific ranges and depths. However, the received signal squared pressures are similarly influenced by the bandwidth, frequency response function, and the vertical beam response function.

To aid the user, many of the model inputs have been initialized in data statements. This eliminates the repeated input of variables which are common to many situations and yet allows the initial values to be suppressed in the event that different values are desired. Those parameters which have been initialized are given in the model input section of the program listing, Appendix C.

MODEL CALIBRATION PROCEDURE

The functional relationships used for noise express the pressure spectrum levels one would expect to measure and are not source levels. To get the effective source levels at a distance of 1 yard (for unit surface area) for distributions of noise generators requires removing from the original levels the effects of frequency-dependent attenuation in the medium and then normalizing to a unit of surface area. This is accomplished by distributing the noise generators along the horizontal noise plane and propagating the noise at different frequencies. A calibration function can then be obtained which is added to the original levels to yield corrected levels and eliminate bias. This effectively removes the effects of frequency attenuation in the original levels and converts them to source levels for unit surface area. Finally, the output of the model is adjusted to coincide with measured data for one set of conditions (frequency, depth, wind speed, etc.) at one location.

This calibration is dependent on the manner in which the noise sources are distributed and the method by which the noise generation area is calculated and is independent of the medium. Hence, once this calibration is performed for one set of conditions at one location, it need not be done for other conditions or locations. The initial calibration was performed using Marine Physical Laboratory (MPL) Pacific ambient noise data (Ref. 11) and shipping information provided by Western Seas Frontier.

MODEL VALIDATION

The final step, validation, has not yet been completed. This requires comparing the model output with measured data for many different locations, seasons, depths, frequencies, shipping distributions, wind speeds, etc. Tentative plans include comparing with the IOMEDX data, which were taken in the Mediterranean during November 1971. Comparison with other data sets for different conditions will be done as data become available. Although a large quantity of data is now readily available, the lack of shipping information for the time interval during which the measurements were taken disqualifies it. Nearly all historical ambient noise data are inappropriate for model validation.

DISCUSSION OF RESULTS

GENERAL RESULTS

Examples of output from the model for three different geographic locations (Fig. 5) have been included to illustrate both the capability of the model and variability in the directional character of the noise field resulting from differences in sensor depth and acoustic propagation (Figs. 6, 7, and 8). Profiles which generally characterize summer conditions in the Pacific, Atlantic, and Mediterranean were chosen.

It is interesting to note the existence of the trough (low level of noise) in the noise field near the horizontal. This condition results from the horizontal rays at the hydrophone

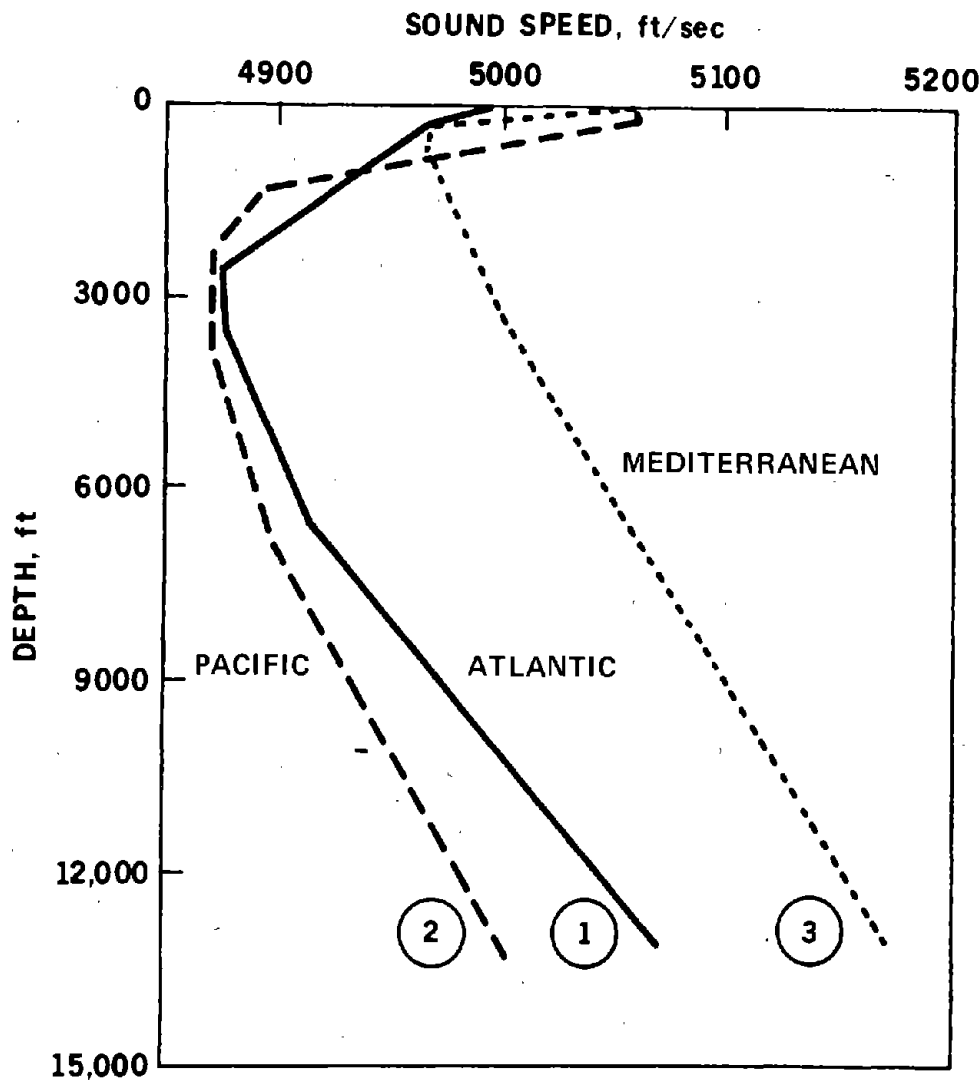


Figure 5. Sound-speed profiles for Areas 1, 2, and 3.

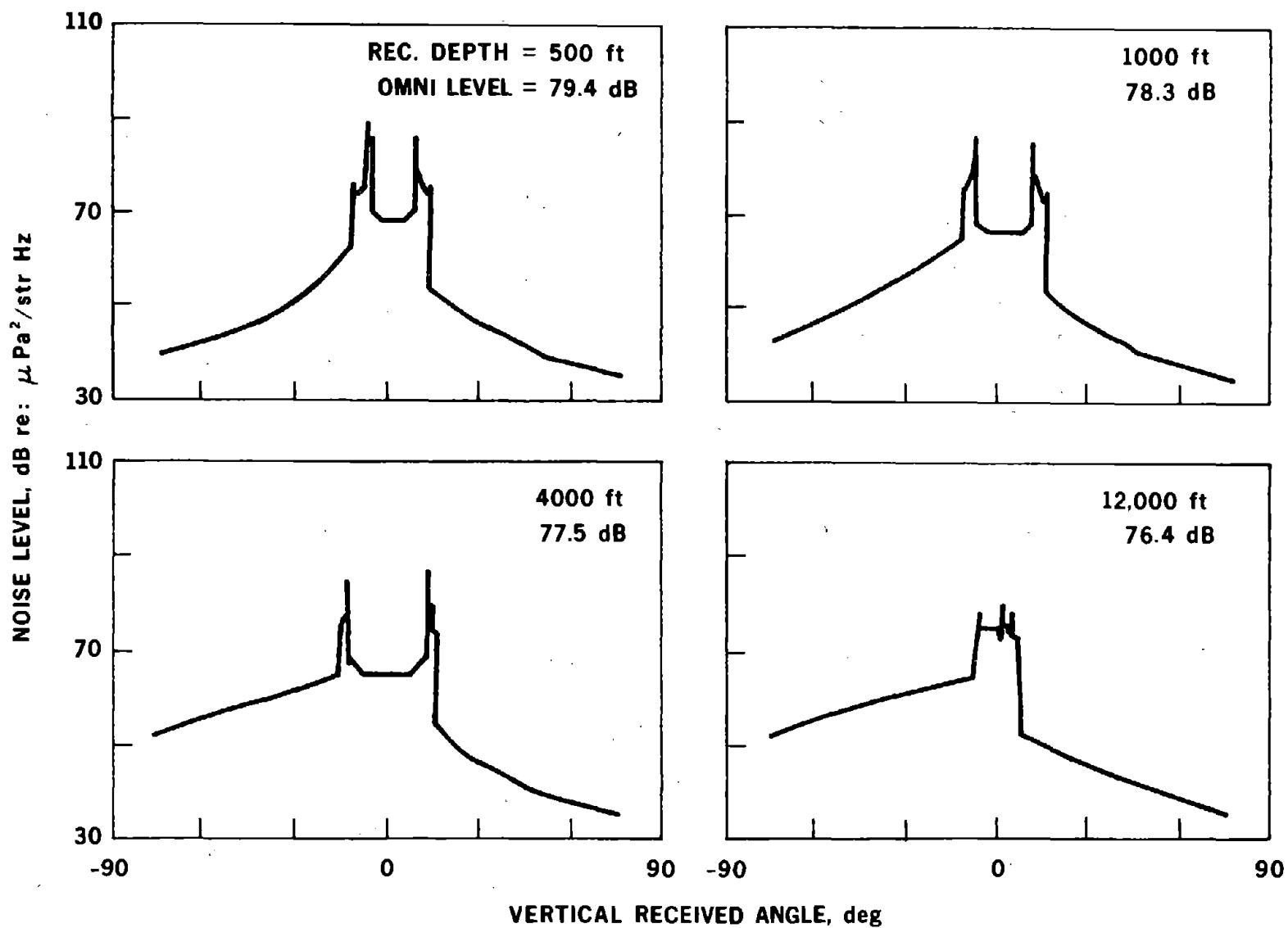


Figure 6. Vertical directionality of ambient noise at four different depths for Area 1.

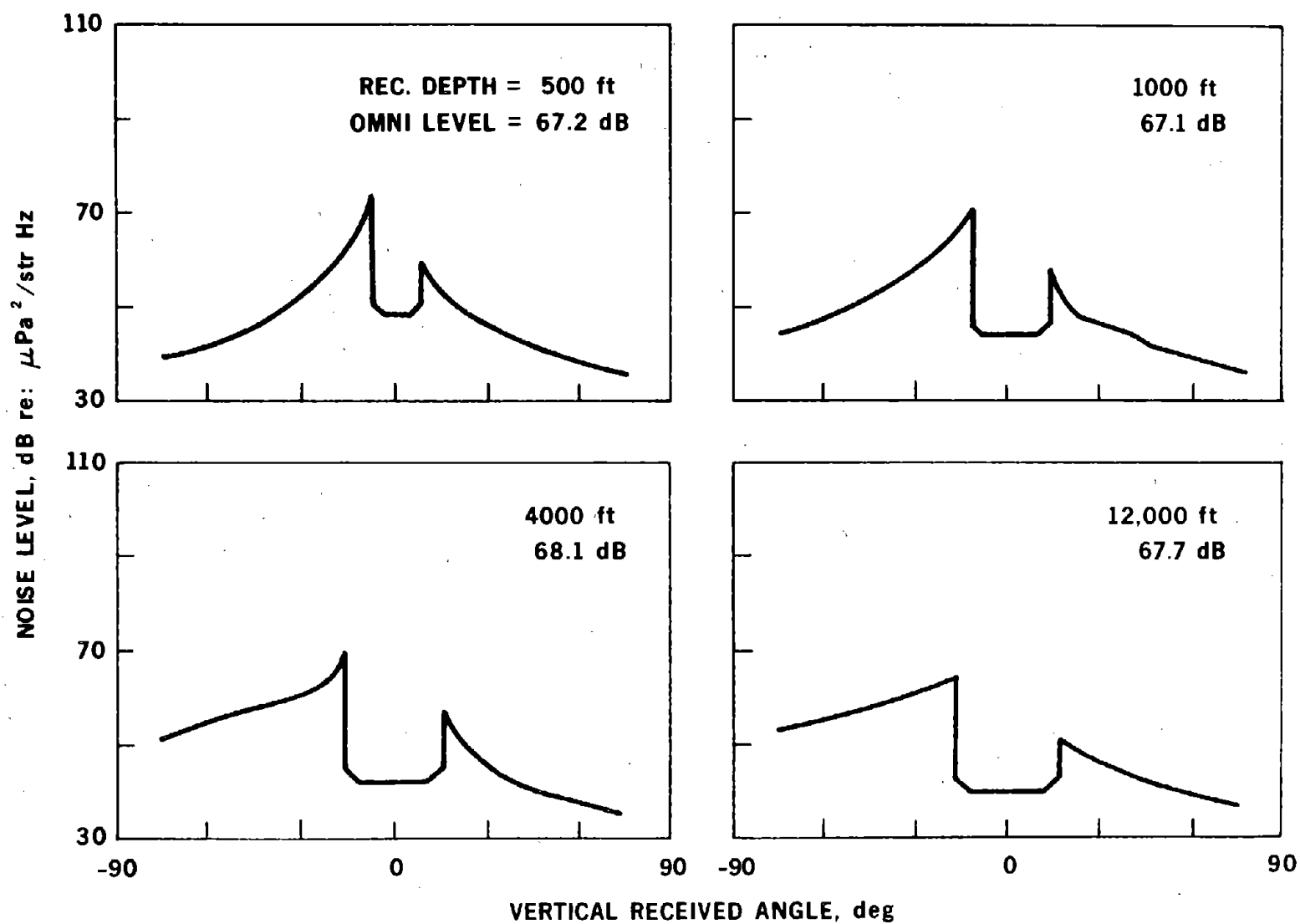


Figure 7. Vertical directionality of ambient noise at four different depths for Area 2.

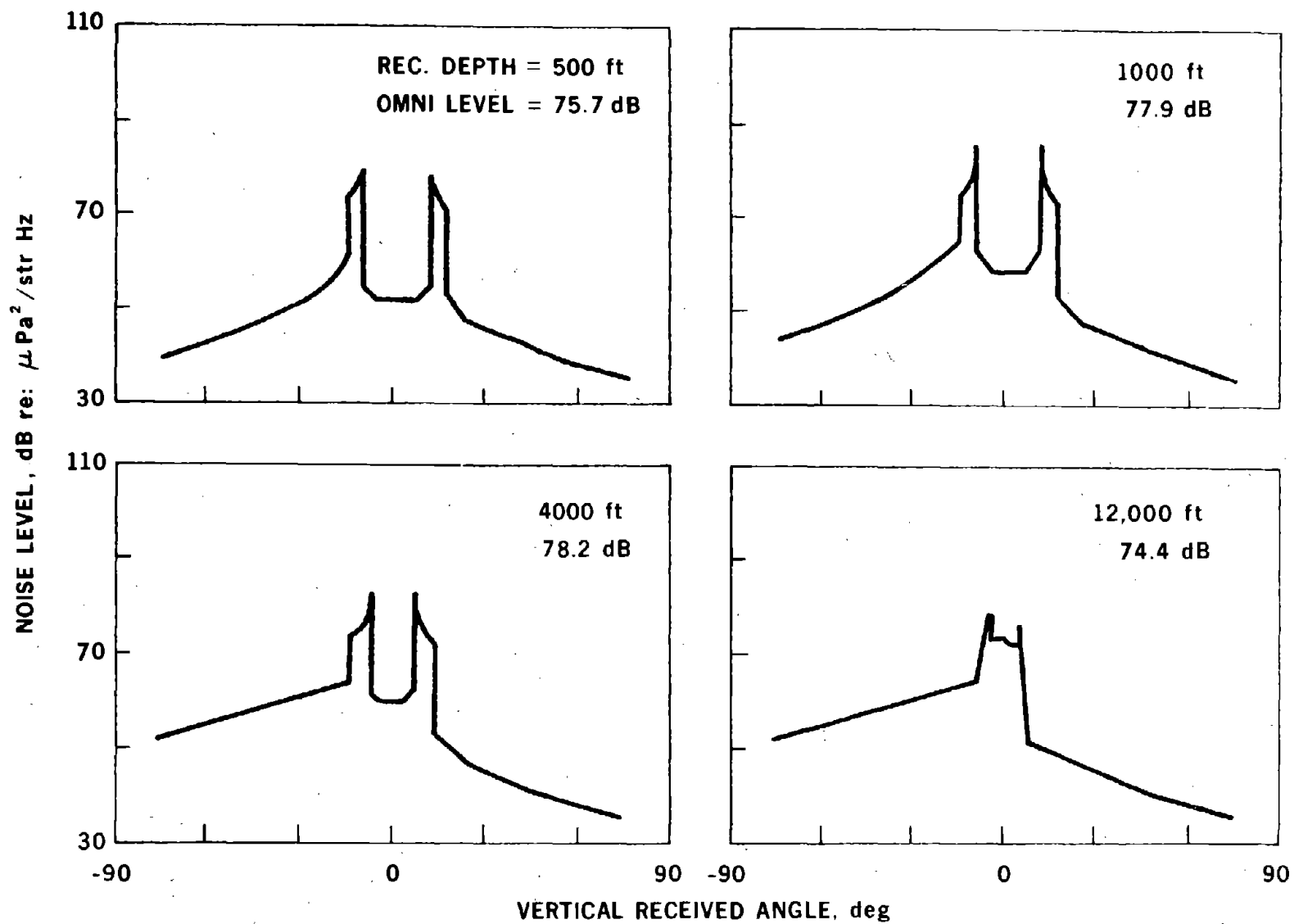


Figure 8. Vertical directionality of ambient noise at four different depths for Area 3.

being unable to reach the surface, where the wind and shipping noise is generated. In two cases, as the depth increases, the trough becomes narrower and eventually disappears, indicating that only near the bottom will the horizontal rays "see" the surface. In the other case, however, the trough gets wider as the depth increases. This results from the sound speed at the bottom being much less than that at the surface; hence, horizontal bottom rays can never reach the surface. Energy within the trough arrives from distant sources. The trough levels are determined by the amount of energy which has become trapped within the SOFAR Channel. There can be a trough, or it can be filled in, depending on whether the energy per arrival angle due to near sources is greater or less than that due to distant sources. The total amount of sound energy received from the distant sources depends on the total number of ships over the continental slope and shelf, range to the shelf, and the total channel "look" angle at the sensor depth.

The lower level of sound at down going angles, greater than approximately 15 deg, compared to the up going angles, less than -15 deg, results from bottom loss. The upcoming rays, having bounced off the bottom before reaching the receiver, suffer a loss in energy not experienced by rays coming from the surface. This accounts for the asymmetric shape of the noise plots.

MODEL OUTPUT-MEASURED DATA COMPARISON

Ambient noise data obtained by MPL (Ref. 11) during their April and May 1971 FLIP ambient noise measurements were utilized for the initial model calibration and to perform a level-variable dependence check.

The model was initially calibrated at 50 Hz and 100 m receiver depth by reading in the measured wind speed and recorded shipping data and adjusting the model output to correspond to the measured noise level for a single day of the 15-day series of measurements. The output of the model was then compared with the measured data for other frequencies, depths, wind speeds, and shipping distributions.

Figures 9a and b are examples of how RANDI compares with the MPL data as a function of time. Only every other point of the original MPL data (one each day) has been retained to compare with the model, since the shipping distribution near the FLIP site was reported only once each day. Figure 9a is a comparison of the MPL data and the RANDI output for 200 Hz at 560 m receiver depth. This frequency is of particular interest in checking the output of the model, since, at this frequency, both shipping and wind-wave interaction make significant contributions to the total noise level. This figure indicates that the model is, in general, within 2 or 3 dB of the measured data. The anomalous results can be explained by the presence or lack of nearby (within the same 1-deg square as FLIP) shipping which was reported (erroneously perhaps) but was not seen by FLIP personnel or could have been close to FLIP's position but not reported. The agreement at 400 Hz (Fig. 9b) is similarly encouraging for such a rough initial calibration.

Figure 9c compares the noise levels calculated for MPL by CAPT Paul Wolff (Ref. 11) using two different noise models, the noise as calculated by RANDI and the

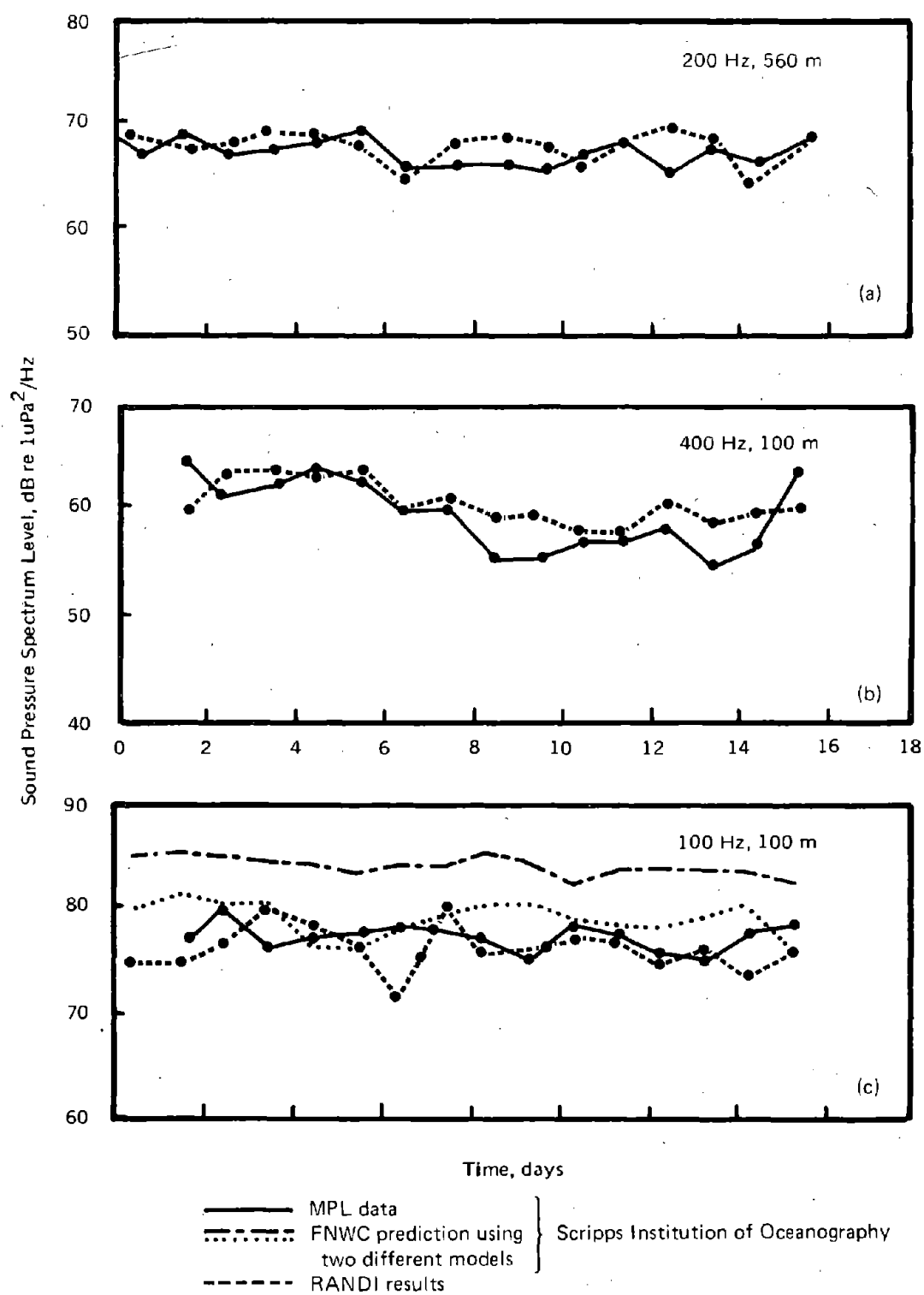


Figure 9. RANDI model-measured data comparison for several frequencies and depths (Ref. 11).

MPL data for 100 Hz at 100 m depth. Again the agreement between RANDI and the MPL data is good.

RANDI also indicated that the noise field should have a depth dependence. The total reduction in noise level as the sensor depth increased from 100 to 3930 m was calculated to be 6 dB at 50 Hz and 4 dB at 400 Hz. The data, however, showed a reduction of about 10 and 4 dB, respectively, for the same conditions. The discrepancy at 50 Hz is largely attributable to not specifying distant (SOFAR Channel) noise sources in RANDI when the comparisons were made. This could easily add the required 4 dB, since the total SOFAR Channel "look angle" at the axis is approximately 22 deg and the nearby shipping was relatively light. A similar depth dependence was observed by Lomask and Frassetto in the Mediterranean, where they utilized the bathyscaph *Trieste* (Ref. 12). Weigle and Watt (Ref. 13), however, generally observed less than 3 dB depth dependence in the Mediterranean for the same frequency range. The environmental and shipping data from Ref. 13 were used as input for RANDI to enable comparison with the measured data. The results are presented in Table 1. Since the data are classified, only the differences in the measured and calculated (by RANDI) levels are given as a function of frequency and depth. It is interesting to note that although RANDI was calibrated with Pacific data, where the ambient noise levels are about 20 dB less than in the Mediterranean, RANDI calculated the Mediterranean noise within about 1 or 2 dB.

RANDI was also used to obtain the horizontal directionality of the ambient noise for one shipping distribution and wind speed during the FLIP ambient noise measurement program. The results are given in Figs. B-6 through B-11.

FUTURE IMPROVEMENTS

Improvements in the current ambient noise directionality model will be made as time and state-of-the-art permit.

Transient noise, including noise from seismic disturbances and explosions, is of considerable interest and will be included in RANDI within the immediate future.

SUMMARY

RANDI is a working FORTRAN model which calculates and plots the directional character of low-frequency ambient noise in the ocean environment. This model is conceptually simple and will be improved as time and state-of-the-art permit. It is intended to be a tool to aid in the design of noise measurement experiments and in surveillance systems design and evaluation and not for ambient noise synoptic forecasting.

Table 1. Comparison of RANDI Output and Measured
Noise Pressure Spectrum Level Data From the
Mediterranean (Ref. 13).

Frequency, Hz	Depth, ft	Difference in Level, dB
20	925	-1.0
	1728	-0.6
	3785	...
	5795	-1.9
	6730	-1.8
50	925	1.9
	1728	2.6
	3785	0.4
	5795	2.7
	6730	4.5
100	925	0.5
	1728	0.0
	3785	0.3
	5795	1.1
	6730	2.4
200	925	-1.2
	1728	-1.6
	3785	-2.8
	5795	-0.5
	6730	-0.1
400	925	0.2
	1728	0.3
	3785	-1.6
	5795	-1.6
	6730	-0.4
500	925	0.5
	1728	-0.7
	3785	-0.5
	5795	-1.6
	6730	-1.1

REFERENCES

1. Navy Electronics Laboratory. Technical Memorandum 163, A Note on the Near Surface Directionality of Deep Sea Noise at 1,000 CPS, by C. N. Miller, 8 February 1956.
2. Naval Research Laboratory. Report 3796, Some Directional Properties of Deep-Water Ambient Noise, by R. J. Urlick, 16 January 1951.
3. R. J. Talham. "Ambient-Sea-Noise Model," *J. Acoust. Soc. Amer.*, 36:1541-1544, August 1964.
4. Naval Air Development Center. Technical Memorandum 69-008-1, A Model for Ambient Noise Due to Shipping, by C. L. Bartberg, 31 December 1969.
5. Naval Undersea Research and Development Center. Technical Note 649, Active Sonar Performance Prediction Model, by W. H. Watson and R. W. McGirr, November 1971.
6. Naval Oceanographic Office. Special Publication 79, Oceanography for Long Range Sonar Systems. Part I, Introduction to Oceanography and Physics of Underwater Sound in the Sea, by B. K. Swanson, February 1966.
7. Naval Underwater Sound Laboratory. Report 1049, Ambient Noise Spectrum Levels as a Function of Water Depth, by A. J. Perrone, November 1969.
8. Scripps Institution of Oceanography. Ref. 61-4, The Directional Distribution of Ambient Noise in the Ocean, by A. B. Becken, March 1961.
9. David Taylor Model Basin. Report C627, Laboratory Study of the Underwater Sound from the Splashes of a Spray of Water Droplets, by G. J. Franz, July 1955.
10. G. M. Wenz. "Acoustic Ambient Noise in the Ocean: Spectra and Sources." *J. Acoust. Soc. Amer.*, Vol. 34, No. 12:1936-1956, December 1962.
11. Personal communication.
12. M. Lomask and R. Frassetto. "Acoustic Measurements in Deep Water Using the Bathyscaph," *J. Acoust. Soc. Amer.*, Vol. 32, No. 8, August 1960.
13. Personal communication.

Appendix A

RANDI SUBROUTINE DESCRIPTIONS AND FLOW DIAGRAM

Figure A-1 illustrates how the main or executive routine controls the flow of logic through the 21 subroutines which comprise RANDI. This figure can also be used as a guide in "streamlining" RANDI by removing those subroutines which will not be required by the user. In such a case, "dummy" or "empty" subroutines are substituted for the originals. For example, RANDI could easily be loaded into a relatively small computer when the propagation loss versus range is to be read in and no sound-speed profile plot is required. This is accomplished by substituting "empty" subroutines for TARGET, RAPATH, SUMINT, RAYTRC, VERTEX, HSURF, SCLNCE and SSPLOT. If only a printout is desired, PLT could also be deleted.

The following is a brief functional description of each subroutine:

ALFA	contains the equation for the absorption coefficient as a function of frequency and water temperature.
AUXPR	calculates sea surface area intersected by a ray bundle and sums squared sound pressure when propagation loss is read in.
BIO	contains equation for biological noise as a function of frequency, time and biological activity indicator.
BWIDTH	integrates noise or target spectrum using an input bandwidth response function.
ERTHC	corrects the sound speed profile to include the effects of earth curvature.
FUNS	obtains the squared noise pressure from the shipping histogram array for a given range.
FUNU	linearly interpolates between two points.
HLOSS	computes absorption, spreading and refraction losses.
HSURF	computes surface reflection losses.
PLT	plots noise and target levels
PRINTS	prints input and initial variables.

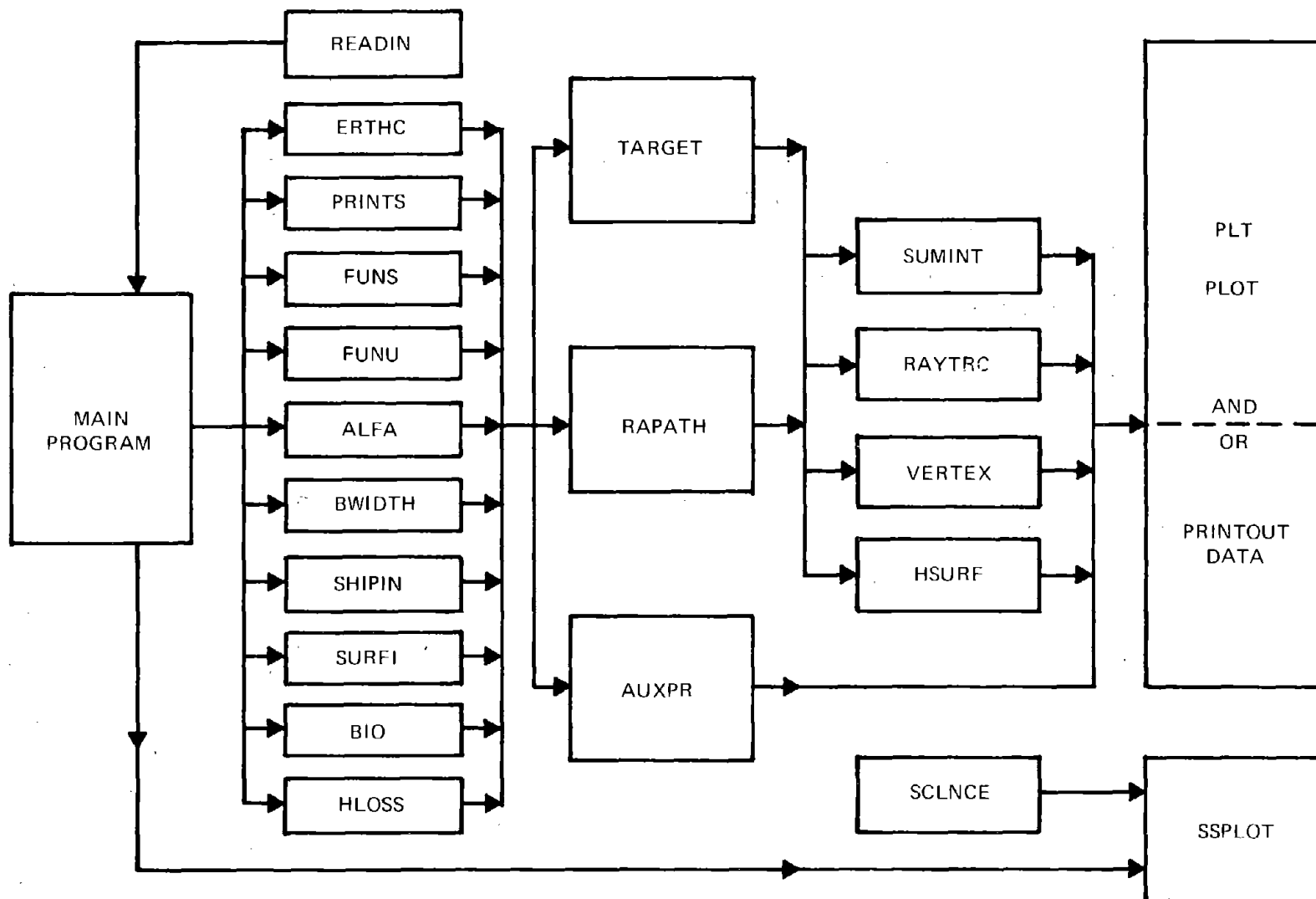


Figure A-1. RANDI subroutine flow diagram.

RAPATH generates ray paths and corresponding propagation losses.

RAYTRC determines path taken by each ray.

READIN reads input parameters.

SCLNCE produces convenient set of scale numbers for sound-speed profile plot.

SHIPIN constructs the ray for the shipping noise squared pressure versus range histogram.

SSPLOT plots sound-speed profile.

SUMINT calculates sea surface area intersected by a ray bundle and sums squared sound pressure when propagation loss not read in.

SURFI contains the squared pressure levels for thermal, shipping, wind-wave and rain noise generators.

TARGET traces rays from target to sensor and computes received signal levels and arrival angles.

VERTEX computes greatest and shallowest depths reached by ray

Appendix B
RANDI INPUT-OUTPUT

RANDI INPUT DATA AND FORMAT

Input data cards for RANDI include parameter cards, array cards and run-control cards.

A RANDI input parameter card consists of an arbitrary number of parameter fields in free format. Each parameter field is separated by a comma, oblique, or blank space and contains the parameter name followed by an equal sign and then the value the parameter is to be assigned for that and subsequent consecutive runs or until a new value is assigned. The array cards contain the name of the array followed by an equal sign, which is immediately followed by the first value in the array. Subsequent values are separated by a blank space. Values which are continued on additional cards are preceded by an asterisk and a blank space at the beginning of the string of values contained on the continuation card.

Data cards for consecutive runs are separated by a PAUSE run control card, which signals the end of data for one run and that further data follows for another run. The last data set is followed by an END-DATA card signaling the end of data for the last run.

The PRLOSS card precedes each array of propagation loss versus range and follows the parameter and array cards for a given run but precedes the PAUSE card. The plot data cards follow the PAUSE card at the end of that data set for a given run, since they are read directly by the plot routines and not the READIN subroutine.

Although it is possible to read in a total of thirty-six parameters and arrays, RANDI can be run, for the large majority of cases, by merely reading in a few variables. For example, if only the sound-speed profile, frequency, and sensor depth are read in, RANDI will use preprogrammed values for the other variables and arrays in calculating the ambient noise. The preprogrammed values are listed in the model input section of the model listing of Appendix C. The resulting noise calculation, then, will be for the given depth, a wind speed of 5 knots, moderately heavy shipping, an omnidirectional vertical beam response function, and a 1-Hz bandwidth centered at the input frequency integrated over a bandpass frequency response function. A 360-deg horizontal sector width, moderate bottom loss, and a latitude of 40-deg will also be used.

The following is a list and brief description of the RANDI input variables and output products.

INPUTS

System

<u>Name</u>	<u>Units</u>	
ADANF	ARRAY	Frequency (kHz) and depth (ft) array
ZTG	FEET	Noise source depth
PHID	DEG	Declination-elevation angle

<u>Name</u>	<u>Units</u>	
DELPH	DEG	Vertical half-power beamwidth
DGREH	DEG	Width of horizontal sector
BERNG	DEG	Bearing of horizontal sector
BNCS		Number of ray bounces
ABW	ARRAY	Frequency response (dB down) across bandwidth
ARESP	ARRAY	Vertical beam response (dB down) pattern

ENVIRONMENTAL

ALAT	DEG	Latitude for earth curvature effects
AVELP	ARRAY	Sound-speed (ft/sec) vs depth (ft) profile
AHB	ARRAY	Bottom reflection loss (dB) vs grazing angle (deg)
APROP	ARRAY	Optional prop loss (dB) vs range (kyd) with elevation angle (deg) and ray bundle width (deg)

NOISE SOURCES

WSPD	KT	Wind speed in knots
ACTIV		Biological activity
HOURL	HR	Local time of day in hours (military)
RAIN	IN/HR	Rainfall
SHIPD		Ship density scale
ASHIP	ARRAY	Ship density (No./10000 sq mi) vs range (nmi)
SHLFR	NM	Average distance to continental shelf and surfaced sound channel
SHLFS		Number of ships within horizontal sector over continental shelf and in surfaced sound channel

TARGET

ZTG1	FT	Target depth
ZTGNO		Number of target depths
ZTING	FT	Target depth increment
RGT	KYD	Range to target
RGTNO		Number of target ranges
RGINC	KYD	Target range increment
TARG		If less than 0 TARG0 is a target line component in dB/micropascal, otherwise TARG0 is const in target spectrum level equation
TARG0		Level of line component in dB or const in spectrum equation
TARG1		Coefficient of $\log f$
TARG2		Coefficient of $(\log f)^2$
TARG3		Coefficient of $(\log f)^3$

OUTPUT AND PLOT CONTROL CARDS

System

Name

OUTPT	Output data control parameter
SNPLT	Noise plot flag
HFLAG	Noise plot data card flag
SSPLT	Sound-speed profile plot flag

Three additional cards required after pause for noise plot

Title

Location

Date

Two additional cards required for sound-speed plot

Location

Date

INPUT DATA CONTROL CARDS

HEADER	Used before a message statement
PAUSE	Used before each consecutive run except first
PRLOSS	Used before each prop loss array
END-DATA	Used at end of data or before plot cards that are at end of data

OUTPUTS

Plots

1. Noise level versus vertical arrival angle with or without target
2. Sound-speed profile

Table

Noise level versus vertical arrival angle
Target signal level versus vertical arrival angle
Total sector noise level

EXAMPLE COMPUTER RUN

Input

An example of a typical RANDI run follows to illustrate the input format and output products. In this example both a vertical directionality plot and a sound-speed profile

plot are requested. The input variables are sensor depth, frequency, latitude, wind speed, request for both plots, bottom loss versus grazing angle array and sound speed versus depth array. Three plot data cards are also included, giving the title, location, and date. All other parameters are automatically set equal to preprogrammed initial values.

Output

Figures B-2 through B-5 illustrate the output of RANDI for a given set of input data cards.

EXAMPLE OF HORIZONTAL DIRECTIONALITY

Shipping data for May 1, 1971, within a few hundred miles of the FLIP ambient noise measurement site, were utilized to illustrate the horizontal directionality capability of RANDI. To do this the noise field was divided into 36 horizontal pie-shaped sectors centered at the FLIP position. The vertical arrival structure of the noise field was obtained by running RANDI with different shipping distributions each time to get the average noise level every 10 deg. These levels were then reduced by 10 dB to get the per degree levels and plotted by a separate DISSPLA plot routine. Four of the 10-deg-horizontal-sector vertical directionality plots (Figs. B-8-11) have been included to demonstrate the variability in the vertical arrival structure.

AVFLP=0 5033 328 5033 656 5000 1640 4987 3280 4902 4920 4898 9514 4957
 * 19685 5111
 AHR=0 0 18 0 38 4 55 6 90 6
 ASHIP=93 1.9 -12 -300 154 5 -290 215 2.9 -12 277 2.1 -11.5 337
 ADANF=.05 975
 ALAT=24.WSPD=5.SNPI T=1.SSPIT=1.SHI FS=50.0UTPT=1
 HEADER
 EXAMPLE PLOTS
 END-DATA

** INITIAL PARAMETERS **

FREQ = .050 KHZ	ZY = 975.0 FT	ZTG = 20.0 FT	PHID = .0 DEG	DEIPH = 170.0 DEG
RNC5 = 10.0	WSPD = 5.0 KT	DGRFH = 360.0 DEG	REFRG = 270.0 DEG	HOUR = 1500.00
ACTIV = .0	RATN = .0 IN/HR	ALAT = 24.0 DEG	SNPI T = 1.0	SSPI T = 1.0
HFLAG = .0	ZTG1 = -1.0 FT	ZTGNO = 1.0	ZTINC = .0 FT	RGT = .0 KYD
RGTNO = 1.0	RGINC = .0 KYD	TARG = .0	TARGO = .0	TARG1 = .0
TARG2 = .0	TARG3 = .0	SHTPD = 5.2	SHIFR = 900.0 NMT	SHIFS = 50.0

VFL PROFILE		BOTTOM LOSS		BEAM RESP		SHIPPING HISTOGRAM				BANDWIDTH	
FT	FT/SEC	DEG	DB	DEG	DB	NMT	SHTPS	KT	FT	HZ	DB DOWN
.0	5033.0	.0	.0	-90.0	.0	93.0	1.9	12.0	300.	.0	.0
328.0	5033.0	18.0	.0	90.0	.0	154.0	5.0		290.	1.0	.0
656.0	5000.0	38.0	4.0			215.0	2.9	12.0			
1640.0	4987.0	55.0	6.0			277.0	2.1	11.5			
3280.0	4902.0	90.0	6.0			337.0					
4920.0	4898.0										
9514.0	4957.0										
19685.0	5111.0										

360.0 DEG SECTOR LEVEL = 40.28 DB FOR 50.0 HZ AT 975.0 FT DEPTH

SOFAR CHANNEL PER STR NOISE LEVEL = 77.0 DB FOR 13.9 VERT DEG TOTAL CHANNEL 'LOOK' ANGLE

Figure B-2. Typical RANDI initial parameter printout.

LOCATION 06100W 2400N
 DATE 03 23 72
 WIND SPEED (KT) 5.0
 BOTTOM DEPTH (FT) 19685.0

FREQ (HZ) 50.0
 REC DEPTH (FT) 975.0
 SECTOR LEVEL (DB) 80.3
 HOR SEC (DEG) 360 AT 270

D/E ANGLE (DEG)	NOISE LEVEL (DB) *	D/E ANGLE (DEG)	NOISE LEVEL (DB) *
-76.5	46.3	76.5	40.4
-72.3	41.0	72.3	39.4
-68.0	44.0	68.0	39.0
-65.9	44.7	65.9	38.8
-63.8	40.4	63.8	38.5
-61.6	44.1	61.6	38.5
-59.5	43.6	59.5	38.4
-57.4	40.5	57.4	38.3
-55.3	43.5	55.3	38.4
-53.1	43.4	53.1	38.5
-51.0	43.3	51.0	38.7
-48.9	43.3	48.9	39.0
-46.8	43.4	46.8	39.3
-44.6	42.0	44.6	39.6
-42.5	43.8	42.5	40.1
-40.4	45.0	40.4	40.5
-39.3	44.9	39.3	40.8
-38.3	44.9	38.3	41.2
-37.2	45.1	37.2	41.5
-36.1	43.7	36.1	41.9
-35.1	46.6	35.1	42.5
-34.0	44.3	34.0	42.8
-32.9	46.8	32.9	43.4
-31.9	43.7	31.9	43.7
-30.8	48.5	30.8	44.5
-29.8	45.4	29.8	44.8
-28.7	47.8	28.7	45.6
-27.6	48.1	27.6	46.2
-26.6	48.0	26.6	46.9
-25.5	49.0	25.5	47.6
-24.4	49.1	24.4	48.4
-23.4	48.7	23.4	49.0
-22.3	51.1	22.3	50.1
-21.3	51.1	21.3	50.8
-20.2	52.0	20.2	51.1
-19.1	60.3	19.1	60.6
-18.1	60.0	18.1	60.5
-17.0	60.7	17.0	61.0
-15.9	64.0	15.9	64.4
-14.9	64.9	14.9	65.1
-13.8	64.4	13.8	65.0
-12.8	67.6	12.8	68.1
-11.8	69.1	11.8	69.3
-10.7	70.4	10.7	70.9
-9.6	73.6	9.6	73.4
-8.5	72.7	8.5	74.3
-7.4	76.9	7.4	77.4

* DB RE MICROPASCAL**2/STERADIAN HZ

Figure B-3. Typical RANDI noise field vertical directionality printout.

EXAMPLE NOISE PLOT

LOCATION 08100N 2400N
DATE 03 23/72
WIND SPEED (KT) 5.0
BOTTOM DEPTH (FT) 19685.0

FREQ (HZ) 50.0
REG DEPTH (FT) 875.0
SECTOR LEVEL (DB) 80.3
HOR SEC (DEG) 360 AT 270

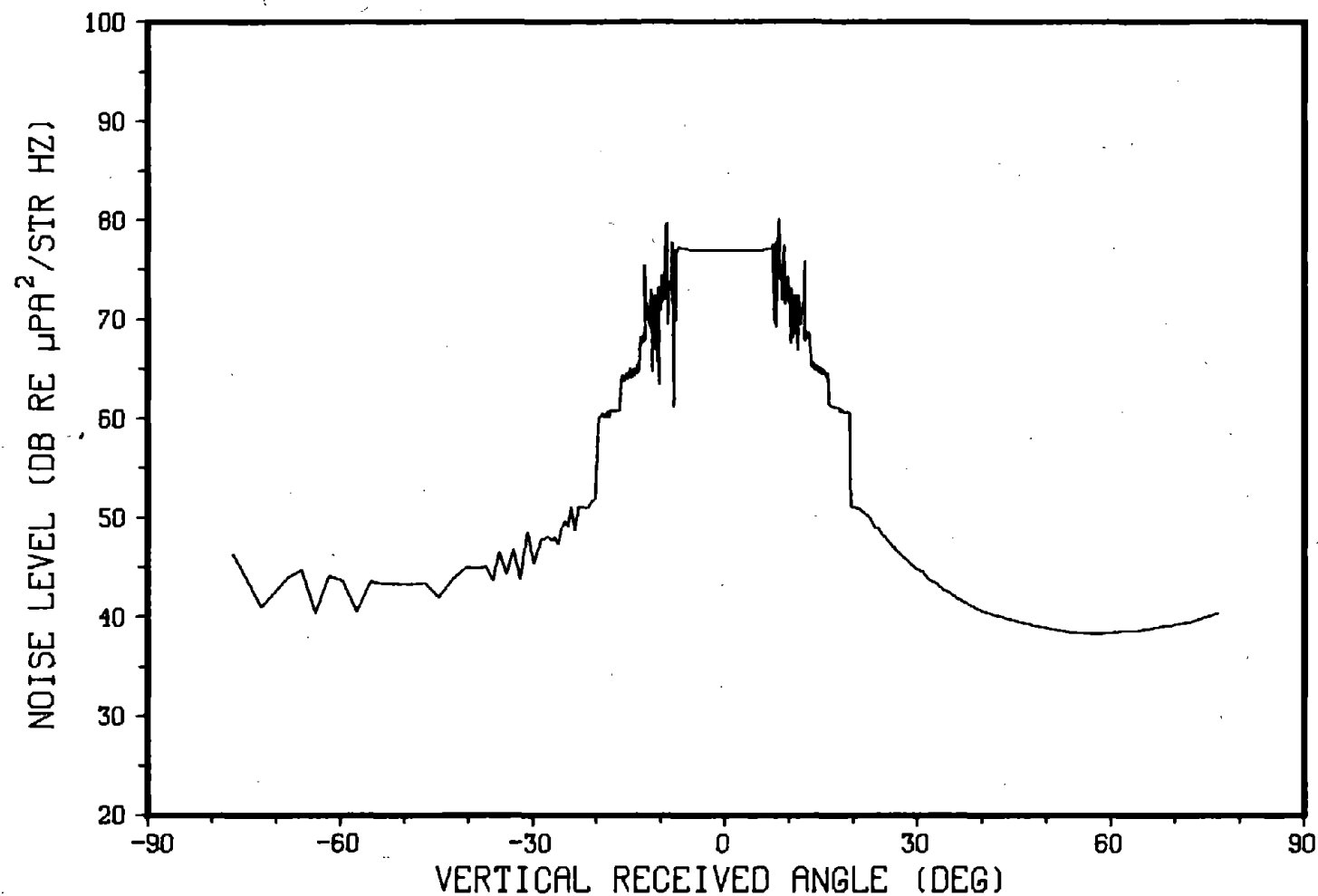


Figure B-4. Typical RANDI vertical directionality plot.

LOCATION 06100W 2400N

DATE 03 23 72

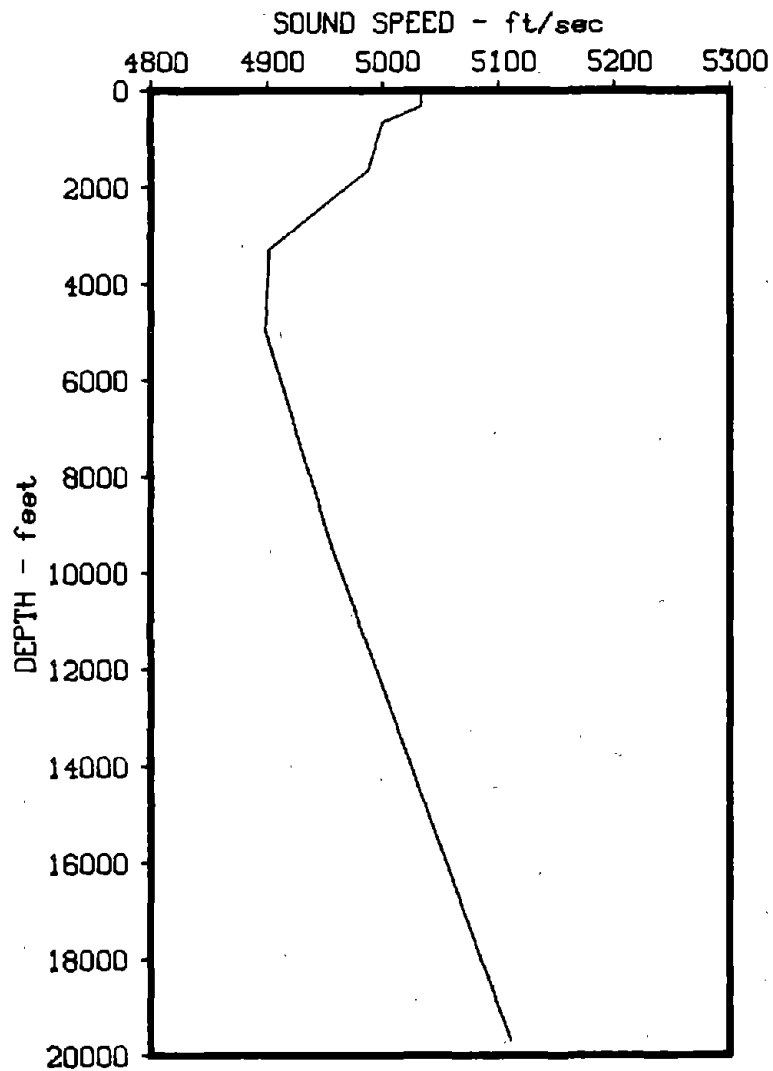


Figure B-5. Typical RANDI sound-speed profile plot.

HORIZONTAL DIRECTIONALITY

NPL PACIFIC DATA - 5/1/71
100 HZ AT 300 FT, 25 KT WIND
80.62 DB OMNI LEVEL

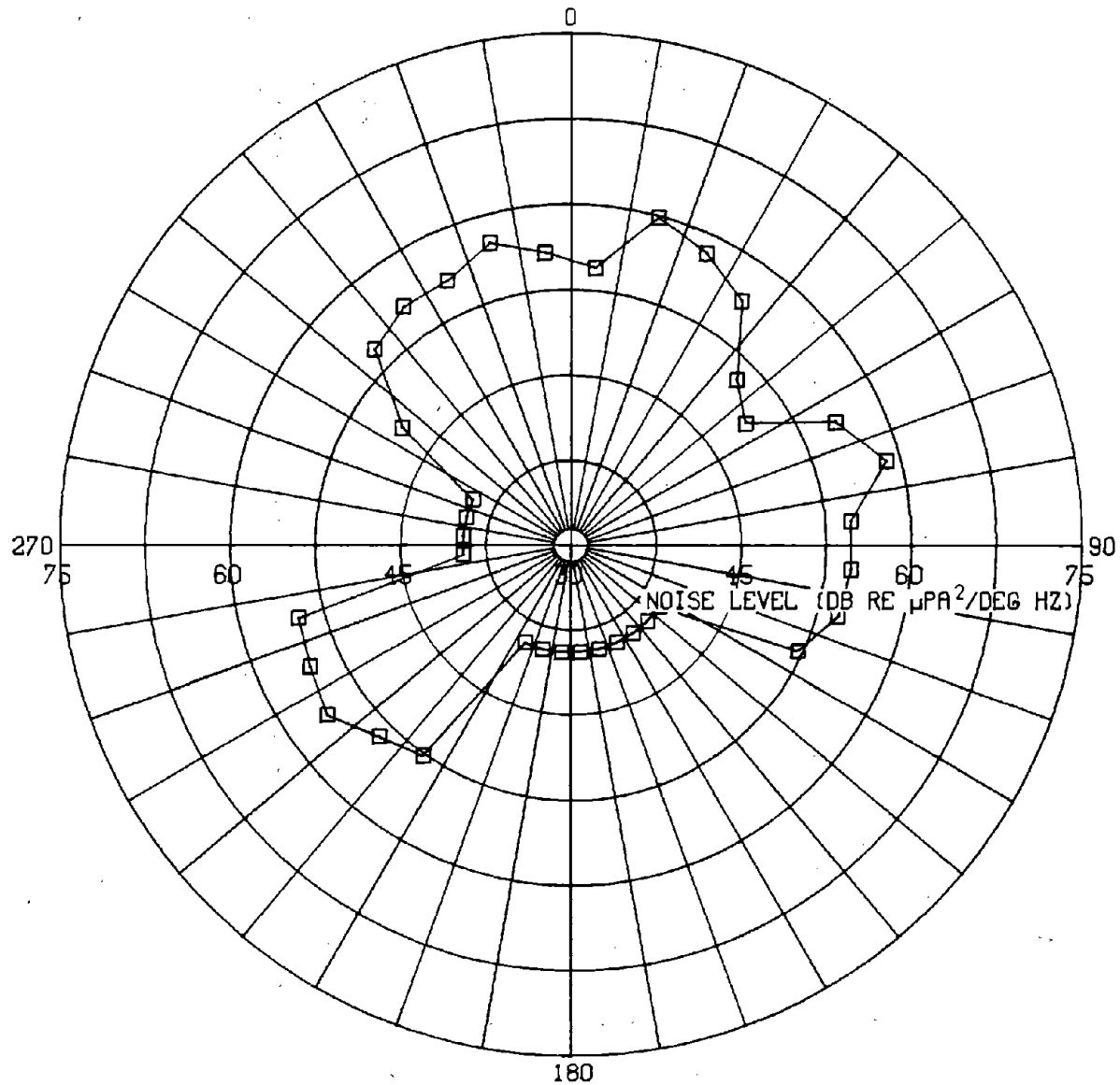


Figure B-6. Per degree horizontal directionality of ambient noise in 10-deg sectors.

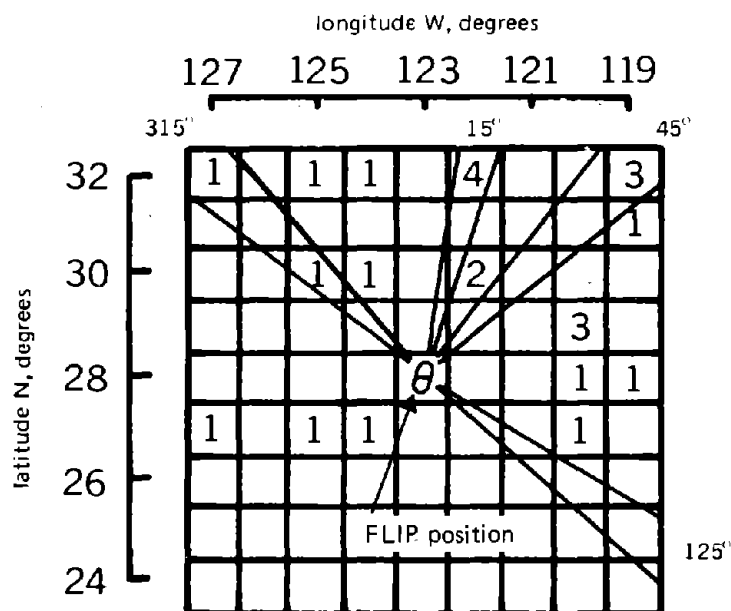


Figure B-7. Pacific shipping data for 1 May 1971. Number of ships over 1000 gross tons per 1-deg square.

MPL DATA COMPARISON

LOCATION 2800N 12300W
DATE 05 01 71
WIND SPEED (KT) 25.0
BOTTOM DEPTH (FT) 13615.0

FREQ (HZ) 100.0
REC DEPTH (FT) 300.0
SECTOR LEVEL (DB) 71.2
HOR SEC (DEG) 10 AT 15

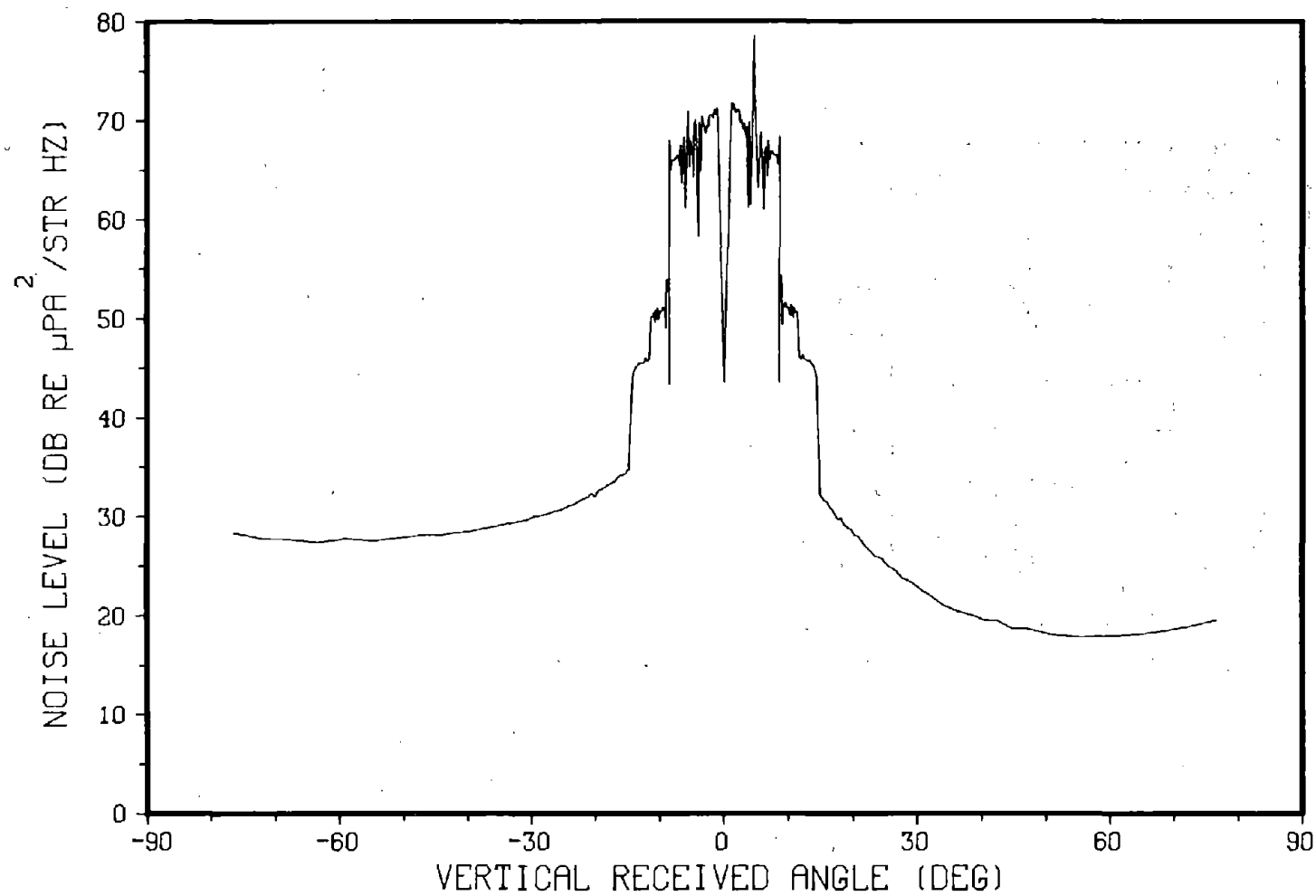


Figure B-8. Ambient noise vertical directionality for the 10-deg horizontal sector at 15 deg.

MPL DATA COMPARISON

LOCATION	2800N 12300W	FREQ (HZ)	100.0
DATE	05 01 71	REC DEPTH (FT)	300.0
WIND SPEED (KT)	25.0	SECTOR LEVEL (DB)	66.2
BOTTOM DEPTH (FT)	13615.0	HOR SEC (DEG)	10 AT 45

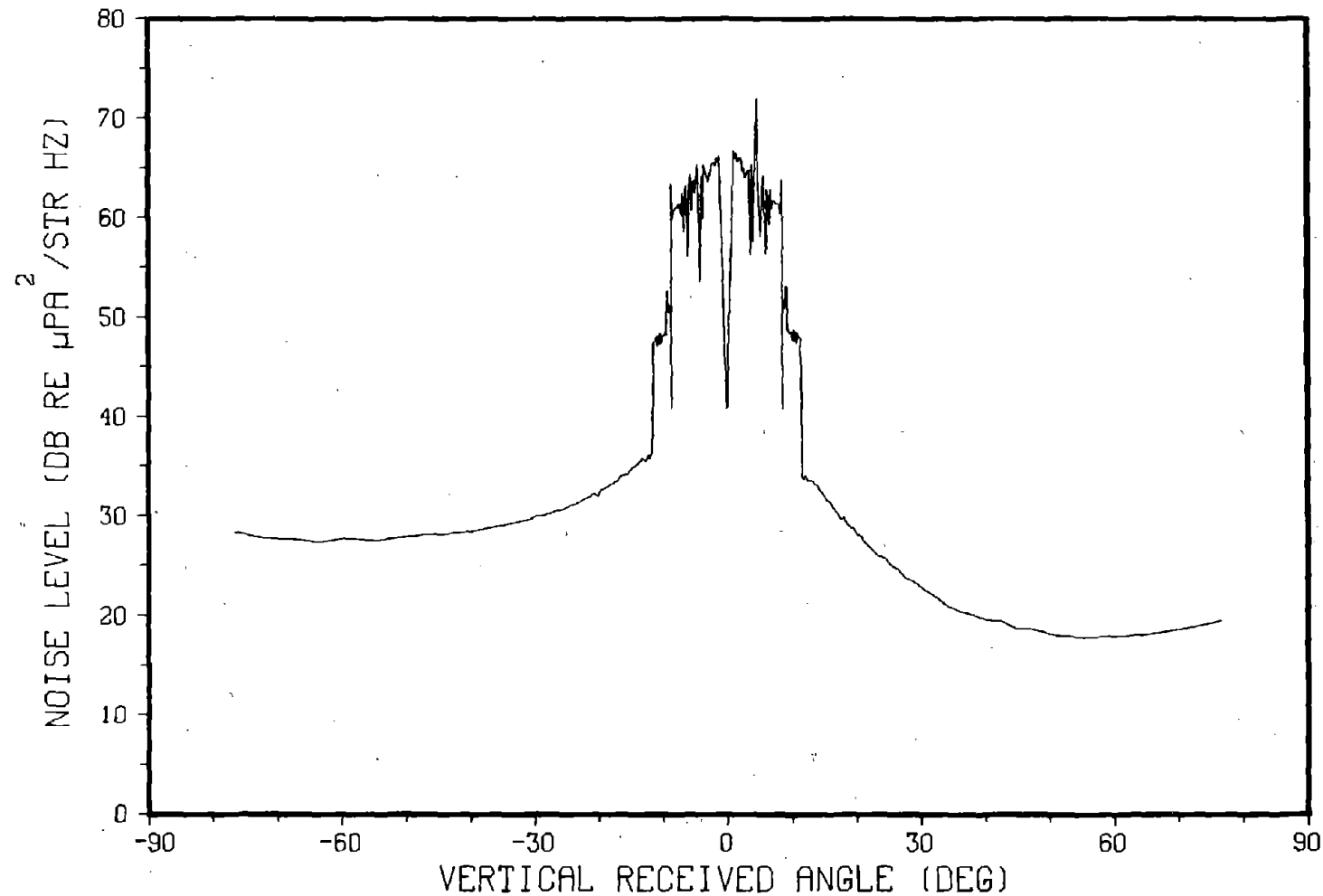


Figure B-9. Ambient noise vertical directionality for the 10-deg horizontal sector at 45 deg.

MPL DATA COMPARISON

LOCATION 2800N 12300W
DATE 05 01 71
WIND SPEED (KT) 25.0
BOTTOM DEPTH (FT) 13615.0

FREQ (HZ) 100.0
REC DEPTH (FT) 300.0
SECTOR LEVEL (DB) 46.5
HOR SEC (DEG) 10 AT 125

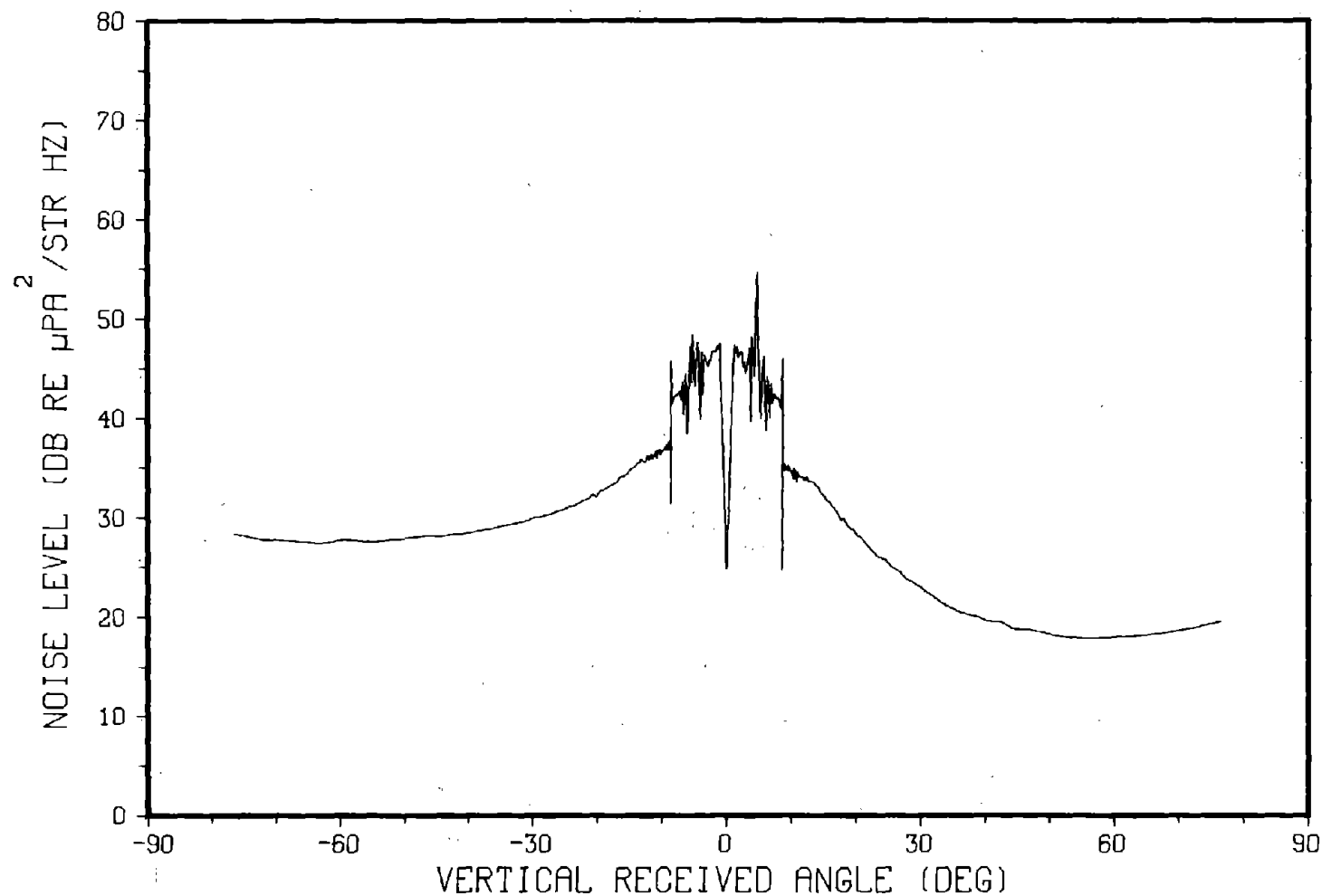


Figure B-10. Ambient noise vertical directionality for the 10-deg horizontal sector at 125 deg.

MPL DATA COMPARISON

LOCATION 2800N 12300W
DATE 05 01 71
WIND SPEED (KT) 25.0
BOTTOM DEPTH (FT) 13615.0

FREQ (HZ) 100.0
REC DEPTH (FT) 300.0
SECTOR LEVEL (DB) 65.6
HOR SEC (DEG) 10 AT 315

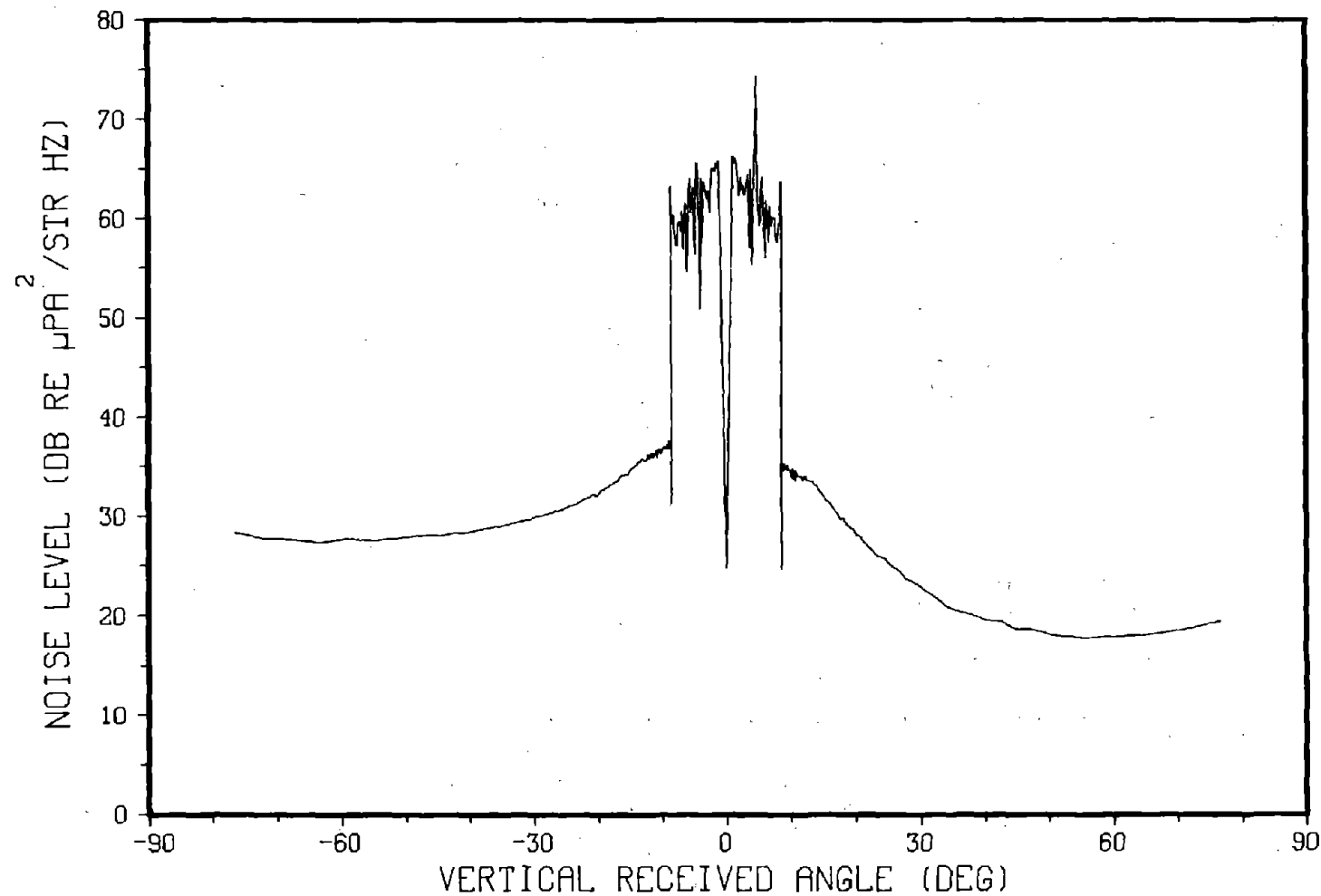


Figure B-11. Ambient noise vertical directionality for the 10-deg horizontal sector at 315 deg.

TARGET - NOISE PLOT EXAMPLE

LOCATION 08100N 2400N
 DATE 03 23 72
 WIND SPEED (KT) 5.0
 BOTTOM DEPTH (FT) 19885.0

FREQ (HZ) 50.0
 REC DEPTH (FT) 975.0
 SECTOR LEVEL (DB) 80.3
 HOR SEC (DEG) 360 AT 270

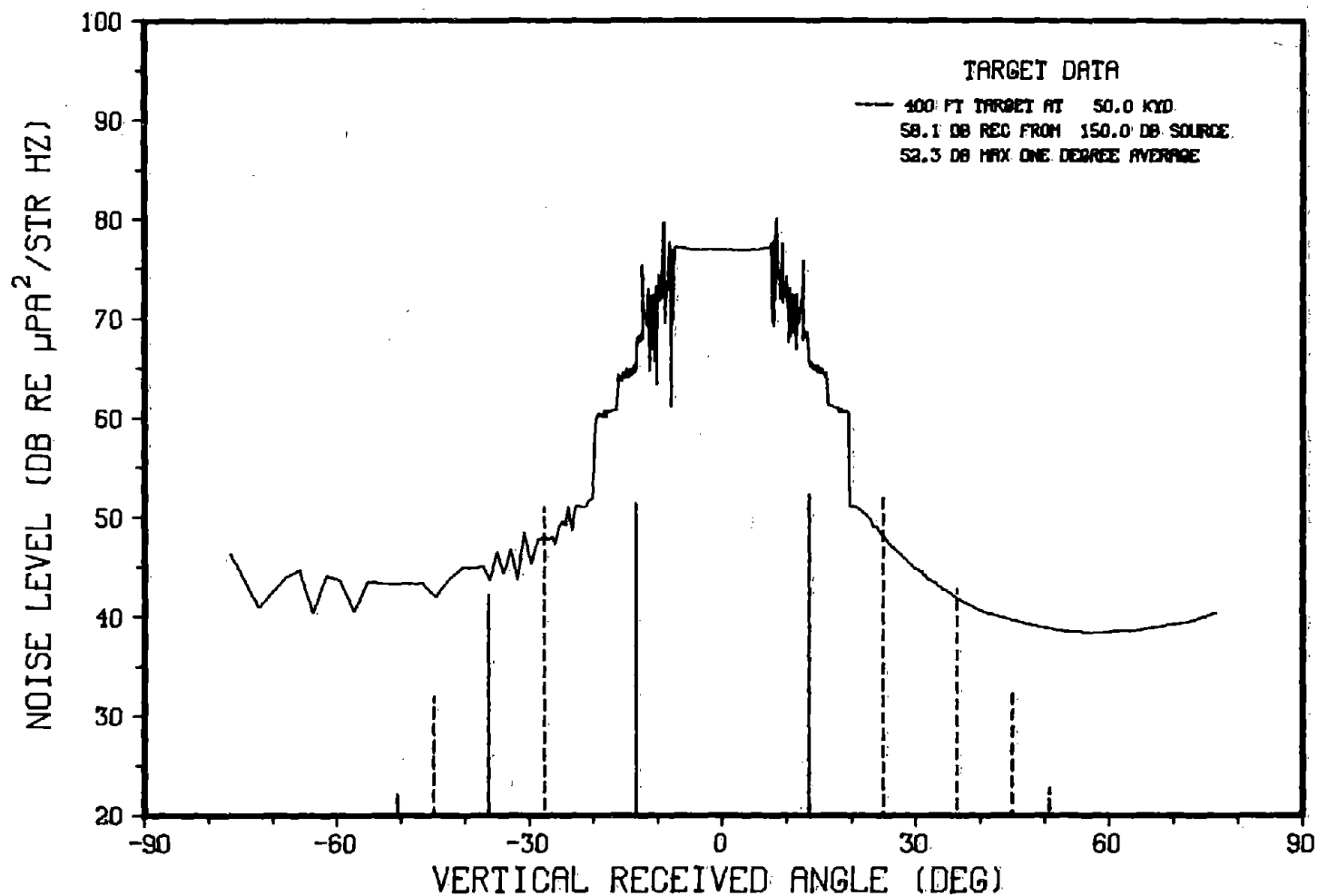


Figure B-12. Example target-ambient noise plot.

REC ANGLE DFG	TARGET LEVEL DB/M PA
-82.135	-248.50
-79.270	-140.83
-76.405	-78.85
-73.540	-50.95
-70.676	-29.66
-67.811	-15.09
-64.946	-7.88
-62.081	-.58
-59.216	6.88
-56.351	14.42
-50.622	22.13
-44.892	31.94
-36.298	42.19
-27.703	51.15
-13.380	51.42
82.135	-247.99
79.270	-140.30
76.405	-78.33
73.540	-50.41
70.676	-29.11
67.811	-14.55
64.946	-7.33
62.081	-.02
59.216	7.44
56.351	11.99
50.622	22.72
44.892	32.55
36.298	42.84
24.839	52.29
13.380	52.29

58.1 DB SIG REC AT 975. FT FROM 150.0 DB TARGET AT
.0500 KHZ, 400. FT DEPTH, 50.000 KYD

Figure B-13. Example target data output.

Appendix C

**RESEARCH AMBIENT NOISE DIRECTIONALITY
(RANDI) MODEL LISTING**


```

C *****
C *
C * RESEARCH AMBIENT NOISE DIRECTIONALITY (RANDI) MODEL *
C *
C *****
C
C ***** MODEL INPUTS *****
C
C NAME UNITS INITIAL
C
C ** SYSTEM INPUTS **
C
C ADANF ARRAY N FREQUENCIES (KHZ) AND K DEPTHS (FT)
C N+K .LE. 30
C ZTG FEET NOISE SOURCE DEPTH - 20
C PHID DEG D/E ANGLE 0
C DELPH DEG VERTICAL HALF-POWER BEAMWIDTH 170
C DGREH DEG WIDTH OF HORIZONTAL SECTOR 360
C BERN6 DEG BEARING OF HORIZONTAL SECTOR 270
C BNCS BOUNCES .LE. 60 10
C ABW ARRAY FREQUENCY RESPONSE (DB DOWN) ACROSS
C FREQ IN BAND) 1 HZ BW INITIAL
C ARESP ARRAY VERTICAL BEAM RESPONSE PATTERN
C OMNI ASSUMED IF NOT INPUT
C
C ** ENVIRONMENTAL INPUTS **
C
C ALAT DEG LATITUDE FOR EARTH CURVATURE EFFECTS 40
C AVELP ARRAY VELOCITY (FT/SEC) VS DEPTH (FEET) PROFILE
C AHB ARRAY BOT-REFL LOSS (DB) VS GRAZING ANGLE (DEG)
C PROV 5 ASSUMED IF NOT INPUTTED
C APROP ARRAY OPTIONAL PROP LOSS (DB) VS RANGE
C (KYD) ARRAY WITH ELEVATION ANGLE
C (DEG) AND RAY BUNDLE WIDTH (DEG)
C APROP(1) = DEPRESSION ANGLE (DEG)
C APROP(2) = RAY BNDL WDT (DEG)
C APROP(2N+1) = RANGE (KYDS)
C APROP(2N+2) = PROP LOSS (DB)
C
C ** NOISE SOURCE INPUTS **
C
C WSPD KT WIND SPEED IN KNOTS 5
C ACTIV BIOLOGICAL ACTIVITY SCALE 0 TO 10 0
C HOUR HR TIME OF DAY IN HOURS (MILITARY) 1500
C RAIN IN/HR RAIN FALL 0
C SHIPD SHIP DENSITY SCALE 0 TO 7 5.2
C .EQ. 0 - NO SHIPS, = 7 - DENSE
C SHIPPING CALIB - MPL DATA 4/24/71

```

C	ASHP	ARRAY	SHIPPING HISTOGRAM ARRAY - EACH BAR IS	
C			DEFINED BY 2 TO 4 NUMBERS IN THE	
C			FOLLOWING ORDER - RANGE IN NM,	
C			NO. SHIPS PAST THAT RANGE AND	
C			BEFORE NEXT RANGE	
C			NEG AV SHIPS SPEED (KT) (OPTIONAL)	-12
C			NEG AV SHIPS LENGTH (FT) (OPT)	-300
C			LAST DATA POINT IS LAST RANGE	
C	SHLFR	NM	AVERAGE DISTANCE IN NAUTICAL MILES	900
C			TO THE CONTINENTAL SHELF AND	
C			SURFACED SOUND CHANNEL WITHIN THE	
C			SECTOR OF HORIZ ANGLE (DGREH)	
C	SHLFS		NO. SHIPS WITHIN HORIZ SECTOR OVER	0
C			CONT SHELF AND IN SURFACED SOUND	
C			CHANNEL (ATLANTIC VLAM DATA CAL)	

C** TARGET/THREAT INPUTS ** (WHEN PROP LOSS NOT READ IN)

C	ZTG1	FT	TARGET DEPTH .LT. 0 NO TARGET	-1
C	ZTGNO		NUMBER OF TARGET DEPTHS	1
C	ZTINC	FT	TARGET DEPTH INCREMENT	
C	RGT	KYD	RANGE TO TARGET	
C	RGTNO		NUMBER OF TARGET RANGES	1
C	RGINC	KYD	TARGET RANGE INCREMENT	
C	TARG		.LT. 0.0 TARGO IS A TARGET LINE	
C			COMPONENT IN DB/MICROPASCAL	
C			.GE. 0.0 TARGO IS CONST IN TARGET	
C			SPECTRUM LEVEL EQU	
C	TARGO		LEVEL OF LINE COMPONENT IN DB OR	
C			CONST IN SPECTRUM EQU	
C	TARG1		COEFF OF ALOG10(HZ)	
C	TARG2		COEFF OF ALOG10(HZ)**2	
C	TARG3		COEFF OF ALOG10(HZ)**3	

C** OUTPUT AND PLOT CONTROL CARDS **

C	OUTPT		OUTPUT DATA CONTROL PARAMETER	-1
C			.LE. 0 NO VIRT DIR PRINT OUT - ONLY	
C			TOTAL SECTOR LEVEL	
C	SNPLT		.LE. 0 NO NOISE PLOT	0
C	HFLAG		.GE. 0.0 REQ PLOT DATA CARDS	0
C			ELIM PLOT DATA THAT DO NOT CHANGE	
C	SSPLT		.LE. 0 NO VELOCITY PROFILE PLOT	0

THREE ADDITIONAL CARDS REQUIRED AFTER PAUSE FOR NOISE PLOT
(ONLY ONE SET FOR EACH FREQ AND DEPTH LOOP)

C	TYPE	FORMAT	EXAMPLE DATA CARD (IN ORDER)
C	TITLE	40A1	AMBIENT NOISE DIRECTIONALITY'S' NOTE (\$')
C	LOCATION	13A1	4020N 01715E
C	DATE	10A1	07 22 71

TWO ADDITIONAL CARDS REQUIRED FOR SOUND SPEED PLOT

C	LOCATION	13A1	4020N 01715E
C	DATE	10A1	07 22 71

C C C C C C C C C C

```
***** BEGIN PROGRAM *****
```

C

c

C
C

CCC

C

43

```

C
C   STORE FOR PRINT OUT ARRAYS THAT CHANGE
C
C   IF (VFLAG.GT.0.0) GO TO 20
DO 20 I=1,NUMV
  AVEL1(I)=AVFLP(I)
20 CONTINUE
  IF (L3+L4 .EQ. 2) CALL PRINTS
  JCNT=0
  TOTLT=0.0
  ISI=0
  HORZI=1.0E-18
  IF (FREQ.LT..01.OR.FREQ.GT..5) GO TO 170
C
C   MULTIPLE RUN CHECK
C   IF (VFLAG.EQ.0.0) CALL ERTHC
C
C   CALCULATE 11 BANDWIDTH FREQUENCIES
  BANDW=ABW(NUMBW-1)
  FSTRT=.5*(-BANDW+SQRT(BANDW*BANDW+4.0E+6*FREQ*FREQ))
  DO 30 I=1,11
    BB5=I-1
    FRE4(I)=BB5*BANDW*.1
    FRE3(I)=FSTRT+FRE4(I)
    BANGL(I)=FUNU(ABW,FRE4(I),NUMBW)
30 CONTINUE
C
C   DETERMINE BIOLOGICAL SOURCE
C   IF (ACTIV.GT.0.0) CALL BIO
C
C   CALCULATE SURFACE NOISE SOURCE INTENSITIES
C   CALL SURFI
C
C   ALPHAC=ALFA(40.0,FREQ)
C   HORZI=1.0E-18
C   IF(SHLFS .EQ. 0.0) GOTO 32
C
C   CALCULATE INTENSITY/STERADIAN FOR HOR NOISE DUE TO DIST SOURCES
C   CALIBRATE HOR NOISE WITH VLAM DATA (ASSUME 10 SHIPS OVER SHELF
C   AT 600 NM)
C
  RDIST=SHLFR*2000.
  SHLFD=10.*ALOG10(1.+SHLFS)
  RLOS=-86.+15.*ALOG10(RDIST)+ALPHAC*(RDIST*.001)
  DO 31 J=1,11
    BB1=ALOG10(FRE3(J)+6.)-.25
    SL1=-1.*BB1+1.16
    SL2=3.3*BB1-6.274
    CORL=-38.
31 DB(J)=-10.-10.*ALOG10(10.**SL1+10.**SL2)-RLOS+SHLFD+CORL
    CALL BWIDTH(HORZI)
C
C   CONVERT TO INTENSITY/HORIZ DEG
C
  HORZI=HORZI*DGREH*.0027777
32 CONTINUE

  HORUB=10.*ALOG10(HORZI/.109662)+84.8856
C
C   WHICH IS THE HALF ANGLE WIDTH OF SOUND CHANNEL SEEN BY RECEIVER
C   SORSV IS SOUND SPEED AT NOISE GENERATOR DEPTH
C

```

```

CX=FUNU(AVELP,ZX,NUMV)
PHICH=0.0
SORSV=AMIN1(AVELP(2),AVELP(NUMV))
IF(SORSV .GE. CX) PHICH=ACOS(CX/SORSV)*RADCON
C
C SKIP RAYTRACE IF PROP LOSS READ IN
C IF(IPOS .EQ. IPOP) GO TO 175
C
C IF (SNPLT.LE.0.0 .AND. OUTPT .GT. 0.0) PRINT 180
C
C CS=AVELP(2)
C PHIC=0.00001
C
C SEARCH FOR MAXIMUM NEAR SURFACE SOUND SPEED
CM=CS
DO 40 I=1,NUMV,2
IF (AVELP(I).GT.3000.0) GO TO 50
C=AVELP(I+1)
IF (C.GE.CM) CM=C
40 CONTINUE
50 IF (CM.LE.CS) GO TO 60
IF (CX.LT.CM) PHIC=ACOS(CX/CM)+0.00001
60 PHISC=RADCON*PHIC
C
C ZBM=AVELP(NUMV-1)
C CB=AVELP(NUMV)
C
C OBTAIN LIMITING ANGLE FOR BOTTOM REFLECTIONS
C PHILIM=0.0
C IF (CX.LT.CB) PHILIM=ACOS(CX/CB)
C
C INITIALIZE PARAMETERS
C
C PHI LOOP
C
C CALL PRERAY (0,0,0)
C IBB=0
C ICZ=0
C
C OBTAIN INITIAL ANGLE INCREMENT
C PHST=ABS(PHID)+0.5*DELPH
C PHND=PHST-DELPH
C PHINC=DELPH/20.0
C PHI=PHST+PHINC
C PHND=AMAX1(PHISC,PHND)/RADCON
70 PHI=PHI-PHINC
C PHIS=PHI/RADCON
C IF (PHIS.LT.PHND) GO TO 80
C IF (ICZ.EQ.1) GO TO 90
C IF (PHIS.GT.PHILIM) GO TO 90
C IBB=1
C PHIS=PHILIM+.0001
C GO TO 90
80 CONTINUE
C IF(ICZ .EQ. 1) GOTO 100
C
C
C DRAW TROUGH IN NOISE CURVE NEAR HORIZONTAL IF PHICH .GT. .5
C AND GET DISTANT SOURCE (HOR) NOISE COMPONENT

```

```

JCNT=JCNT+1
I51=JP1+1
XANG(I51)=-PHICH*.7
XANG(I51+1)=PHICH*.7
YDB(I51)=10.*ALOG10(10.*(THROB/10.))+10.*(HORDB/10.))+100.
IF(PHICH.LT..5) YDB(I51)=YDB(JP1)
YDB(I51+1)=YDB(I51)
TOTLI=TOTLI+HORZI*2.*PHICH
JP1=JP1+3
C INITIALIZE CZ FOLDING RANGE CHECK
CALL PRERAY (0,0,0)
ICZ=1
IF(PHND.GT.PHILIM) GOTO 110
PHINC=PHILIM-PHIC
IF (PHINC.LE.0.0) GO TO 110
PHINC=-1.4324*PHINC
PHI=(PHIC+0.001)*RADCON
GO TO 70
90 CALL RAPATH (ICZ,PHIS,DELR)
IF (ICZ.EQ.1) GO TO 100
IF (IBB.EQ.1) GO TO 80
IF (DELR.GT.1.) PHINC=0.5*PHINC
GO TO 70
C
100 CONTINUE
IF (PHIS.LT.PHILIM) GO TO 70
110 CONTINUE
C
C CALCULATE OMNI NOISE LEVEL
C
IF (TOTLI.LE.0.0) GO TO 171
TOTLN=10.0*ALOG10(TOTLI)+184.8856
HORDB=HORDB+100.
FRE1=FREQ*1000.0
PRINT 200, DGREH,TOTLN,FRE1,ZX
CANGL=2.*PHICH
IF(SHLFS.GT.0.0.AND.PHICH.GE.0.) PRINT 220,HORDB,CANGL
C
C BYPASS TARGET IF PROP LOSS READ IN
C IF(PTEST.EQ.1.0) GO TO 120
C
IF (ZTG1) 120,130,130
120 IF (SNPLT.GT.0.0) CALL PLT (PTEST)
GO TO 150
C
C TARGET RANGE-DEPTH LOOP
C
130 NT=ZTGNO
NR=RGTNO
HTST=HFLAG
DO 140 I=1,NR
DO 140 J=1,NT
YT=J-1
YR=I-1
RTG=RGT+YR*RGINC
TGDP=ZTG1+YT*ZTINC

CALL TARGET
IF (SNPLT.GT.0.0) CALL PLT (PTEST)
HFLAG=-1.0

```

```

140 CONTINUE
    HFLAG=HTST
150 CONTINUE
C
C    PLOT SOUND SPEED PROFILE IF CALLED FOR
    IF (SSPLT.GT.0.0) CALL SSPLT
C
160 CONTINUE
    HFLAG=HFLG1
    IF (IPOS.EQ.IPOZ) GO TO 10
C
    STOP
C
170 PRINT 210, FREQ
    GO TO 160
171 PRINT 190
    GO TO 150
C    READ PROP LOSS INSTEAD OF CALCULATING IT
175 CALL AUXPR(IPOS,PHICH,HORZI)
    PTEST=1.0
    GO TO 110
C
C
180 FORMAT (//////57X,'OUTPUT'////24X,3HPhi,8X,4HDPHI,8X,5HMAX R,5X,8HLO
    1SS(DB),6X,5HDB UP,6X,7HDB DOWN,3X,10H1 STER COR/)
190 FORMAT (1 TOTLI ZERO OR NEGATIVE - GO TO NEXT CASE')
200 FORMAT (/6X,F7.1,' DEG SECTOR LEVEL = ',F7.2,' DB FOR',F6.1,' HZ
    1AT',F7.1,' FT DEPTH'/)
210 FORMAT (7H FREQ =,F10.4,2BHKHZ OUTSIDE 10-500 HZ REGION)
220 FORMAT (8X,'SOFAR CHANNEL PER STR NOISE LEVEL ='F6.1,' DB FOR'F5.1
    1,36H VERT DEG TOTAL CHANNEL 'LOOK' ANGLE/)
    END

```

```

FUNCTION ALFA (T,F)
C
C    INPUTS
C    T    DEG-F    TEMPERATURE
C    F    KHZ      FREQUENCY
C    ALFA IS THE ABSORPTION COEFFICIENT
    FSQ=F*F
    FCUBE=F*FSQ
    FLOG=ALOG(F)
    TC=5.0*(T-32.0)/9.0
    FT=(6.0*TC+118.0)/(TC+273.0)
    FT=21.9*EXP(2.3026*FT)
    ALFAT=0.0542*EXP(1.5*FLOG)
    FCUT=FCUBE/32.768
    IF (FCUT.LT.1.0) GO TO 10
    BF1=1.0/(1.0+FCUT)
    BF2=1.0-BF1
    GO TO 20
10 BF2=1.0/(1.0+1.0/FCUT)
    BF1=1.0-BF2
20 ALFAM=FSQ/FT*(0.6505317*FT/(FT+FSQ/FT)+0.026847)

    ALFA=BF1*ALFAT+BF2*ALFAM
    RETURN
    END

```

```

C      SUBROUTINE AUXPR(IPOS,PHICH,HORZI)
C
C      THIS SUBROUTINE CALCULATES AMBIENT NOISE WHEN PROP LOSS IS READ
C      IN AND NOT CALCULATED
C
C
C      COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,RNCS,OUTPT,WSPD,DGREH,SNPLT,S
1SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGNO,ZTINC,RGT,RGTNO,R
2GINC,TARG,TARG0,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4,NUMR,NUMS,NUMBW,NUMP,NUMF/COMX/CX,ZBM,PHIC,ALPHAC,BION
C      COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
11,TOTLN,I51,ITST,NTST,THRML,THRDB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC
C
C      DATA IPOZ/'PRLOS.'/
C
C      INITIALIZED PARAMETERS ARE THE FOLLOWING
C
C      TOTLI=1.0E-18
C      TOTLI=TOTLI+HORZI*2.*PHICH
C      JP1=0
C      HORDB = 10.*ALOG10(HORZI)+84.8856
C      IF (PHICH .LT. .5) GO TO 1
C      Q(1)=-PHICH*.7
C      Q(2)=-Q(1)
C      W(1)=10.*ALOG10(10.** (THRDB/10.)+10.** (HORDB/10.))+100.
C      W(2)=W(1)
C      XANG(1)=1.
C      XANG(2)=1.
C      JP1=2
C
C      1 CALL READIN(IPOS)
C
C      CALCULATE WIND NOISE DIRECTIONALITY FACTOR DIRS
C
C      CX=FUNU(AVELP,ZX,NUMV)
C      CS=FUNU(AVELP,ZTG,NUMV)
C      COSPH= COS(.01738*APROP(1))*CS/CX
C      IF(COSPH.GE.1.)COSPH=1.
C      PHI=ACOS(COSPH)
C      EXPN=5.729*PHI
C      IF(PHI.LT..3491)EXPN=2.0
C      DCOEF=2.*FREQ-.02
C      IF(FREQ.GT..5)DCOEF=1.0
C      DIRS=1.+(SIN(PHI)**EXPN-1.)*(DCOEF)
C
C      RI=APROP(3)
C      SHI=FUNS(ASHI1,RI,NMS2)
C      PRI=APROP(4)
C      RECTI=1.0E-18
C      I=-1
C      4 I=I+1
C      NS=2*I+1
C      IF(ASHI1(NS).LE.APROP(3)) GO TO 4
C      NP=5

```



```

C      SELECT RANGE INTEGRATION INTERVAL
C
10  IF (APROP(NP)-ASHI1(NS))30,20,40
20  RTEST=1.0
    NS=NS+2
30  RIP1=APROP(NP)
    PRIP1=APROP(NP+1)
    SHIP1=FUNS(ASHI1,RIP1,NMS2)
    NP=NP+2
    GO TO 50
40  RIP1=ASHI1(NS)
    PRIP1=FUNU(APROP,RIP1,NUMP)
    SHIP1=ASHI1(NS+1)
    NS=NS+2
    IF (RTEST.EQ.1.0) NP=NP+2
50  CONTINUE

C
C      CALCULATE TOTAL SURFACE NOISE INTENSITY DENSITY FUNCTION
C      ONI=A*RANGE*B
C
    RDIF=RIP1-RI
    IF (RDIF.EQ.0.0) GO TO 80
    SDIF=SHIP1-SHI
    A=SDIF/RDIF
    BWIND=WINDI*DIRS
    BSHIP=SHI-RI*A
    B=BWIND+BSHIP
    ONI=SHI+BWIND
    DNIP1=SHIP1+BWIND

C
C      CALCULATE PROP LOSS INTENSITY REDUCTION FACTOR
C      FROM PROP LOSS = E*RANGE + F , WHERE F INCLUDES G THE LOSS
C      DUE TO THE BEAM PATTERN
C
    PDIF=PRIP1-PRI
    E=PDIF/RDIF
    G=FUNU(ARESP,APROP(1),NUMP)
    F=PRI-RI*E-G
    IF (ABS(E) .GE. .0001) GO TO 60
    RECI=6.282*10.**(-.1*F)*(.3333*A*(RIP1**3-RI**3)+.5*B*(RIP1*RIP1
1-RI*RI))
    GO TO 70
60  RECPE=10./(E*2.3025)
    RECI=-6.282*RECPE*(10.**(-.1*PRIP1)* DNIP1*(RIP1+RECPE)-10.**
1(-.1*PRI)*(ONI*(RI+RECPE)))
    IF (ABS(A) .GT. .0001) RECI=RECI-6.282*A*RECPE*RECPE*(10.**(-.1*
1PRIP1)*(RIP1+2.*RECPE)-10.**(-.1*PRI)*(RI+2.*RECPE))

C
C      CALCULATE LEVEL RECEIVED FROM HORIZ SECTOR AND TEST FOR LAST RANGE
C
70  RECTI=RECTI+.0175*DGREH*ABS(RECI)+(THRML+BION)*APROP(2)
80  RI=RIP1

    SHI=SHIP1
    PRI=PRIP1
    IF (NP.LT.NUMP) GO TO 10

C
C      STORE LEVEL AND ANGLE FOR OUTPUT AND GET TOTAL OMNI INTENSITY

```

```

      JP1=JP1+1
      W(JP1)=10.0*ALOG10(RECTI/APROP(2))+184.8856
      Q(JP1)=APROP(1)
      XANG(JP1)=APROP(2)
      TOTLI=TOTLI+RECTI
      IF(IPOS.EQ. IPOZ) GO TO 1
C
C      SORT AND GET MINIMUM LEVEL
C
      JP1M1=JP1-1
100  DO 110 K=1,JP1M1
      L=K+1
      IF(Q(L).GT.Q(K))GO TO 110
      DUMYG=Q(L)
      DUMYW=W(L)
      DUMYX=XANG(L)
      Q(L)=Q(K)
      W(L)=W(K)
      XANG(L)=XANG(K)
      Q(K)=DUMYG
      W(K)=DUMYW
      XANG(K)=DUMYX
110  CONTINUE
      JP1M1=JP1M1-1
      IF(JP1M1.GE.2)GO TO 100
C
C      PRINT OUT IF NO PLOT REQUIRED
C
      IF(SNPLT.GT.0.0)GO TO 130
      PRINT 200
      DO 120 K=1,JP1
120  PRINT 210,Q(K),W(K)
      GO TO 150
130  DO 140 K=1,JP1
140  W(K)=W(K)-10.*ALOG10(XANG(K))
150  CONTINUE
200  FORMAT(/11X,'PHI',4X,'LEVEL (DB)')
210  FORMAT(F15.1,F10.1)
      RETURN
      END

      SUBROUTINE BIO
C
C      THIS SUBROUTINE COMPUTES THE CONTRIBUTION OF BIOLOGICAL NOISE
C
      COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1  SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGNO,ZTINC,RT,RTNO,R
2  GINC,TARG,TARG0,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3  ,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4  ,NUMR,NUMS,NUMBW,NUMP,NUMF/COMX/CX,ZBM,PHIC,ALPHAC,BION
      COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
1  I,TOTLN,I51,ITST,NTST,THRML,THRDB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2  DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
3  30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4  ,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC

```

```

      ACT=ACTIV/5.0
      DO 10 J=1,11
      BB1=ALOG10(FRE3(J))
C
C      B IS THE DAILY FLUCTUATION
C
      B=.00175*(100.-FRE3(J))*SIN(.002618*(HOUR-300.))
      BIO1=-26.6*BB1+20.
      BIO2=28.*BB1-52.64
10  DB(J)=-35.4-10.*ALOG10(10.**(.1*BIO1)+10.**(.1*BIO2))+B*ACT
      1+2.*ACTIV
      CALL BWIDTH (B2)
C
C      CONVERT TO INTENSITY/VERT DEG/HOR DEG
C
      BION=ACT*B2*DGREH*1.5432E-5
C
      RETURN
      END

      SUBROUTINE BWIDTH (DINT)
C
C      THIS SUBROUTINE INTEGRATES THE SIGNAL INTENSITY OVER BANDWIDTH
C      ADD BANDWIDTH RESPONSE TO NOISE SIGNAL
C
      REAL LEVBW(11),MI,MIP1
C
      COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGN0,ZTINC,RGT,RGTN0,R
2GINC,TARG,TARG0,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4,NUMR,NUMS,NUMBW,NUMP,NUMF
      COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
1I,TOTLN,I51,ITST,NTST,THRML,THRDB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TOB(180),TDEG(18
30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC
C
C      DINT=0.0
C
C      OBTAIN EFFECTIVE LEVELS
C
      DO 10 I=1,11
10  LEVBW(I)=DB(I)-BANDL(I)
C
C      INTEGRATE AND CONVERT TO INTENSITIES
C
      DO 30 I=1,10
      FI=FRE3(I)
      FIP1=FRE3(I+1)
      MI=(LEVBW(I+1)-LEVBW(I))/(10.0*ALOG10(FIP1/FI))
      BI=LEVBW(I)-(ALOG10(FI)/ALOG10(FIP1/FI))*(LEVBW(I+1)-LEVBW(I))
      SMB=3.2467E-9*10.**(BI/10.)
      IF (MI.EQ.-1.0) GO TO 20
      MIP1=MI+1.0
      DIN=(SMB/MIP1)*((FIP1**MIP1-FI**MIP1)
      GO TO 30
20  DIN=SMB*ALOG(FIP1/FI)
30  DINT=DINT+DIN

```

```

C      NORMALIZE INTENSITY TO ONE HZ
      DINT=DINT/BANDW

C
      RETURN
      END
      SUBROUTINE ERTHC

C
C      THIS SUBROUTINE CORRECTS FOR EARTH CURVATURE EFFECTS
C
      COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1      SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGN0,ZTINC,RGT,RGTNO,R
2      GINC,TARG,TARG0,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3      ,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4      ,NUMR,NUMS,NUMBW,NUMP,NUMF/COMX/CX,ZBM,PHIC,ALPHAC,BION
      COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
1      1,TOTLN,I51,ITST,NTST,THRML,THROB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2      DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
3      0),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4      ,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC

C
      DATA RADCON/57.2957795/
      DATA AB/43642464.29E7/,ASQ/43789029.00E7/,BSQ/43496390.14E7/

C
C      STORE ORIGINAL BOTTOM DEPTH FOR PLOT DATA
      ZB1=AVELP(NUMV-1)

C
C      EARTH CURVATURE CORRECTION
C
      JPAIRS=NUMV/2
      ALAT=ALAT/RADCON
      SINL=SIN(ALAT)
      COSL=COS(ALAT)
      SINLQ=SINL*SINL
      COSLQ=COSL*COSL
      ROL=AB/SQRT(ASQ*SINLQ+BSQ*COSLQ)
      DO 10 I=1,JPAIRS
        I2=2*I
        AVELP(I2)=AVELP(I2)*ROL/(ROL-AVELP(I2-1))
        AVELP(I2-1)=AVELP(I2-1)*(1.+(AVELP(I2-1)/(2.*ROL))+((AVELP(I2-1)/R
10      10L)**2.)/3.)
10      CONTINUE
      VFLAG=1.0
      ALAT=ALAT*RADCON
      RETURN
      END
      FUNCTION FUNS(A,X,N)

C
C      THIS FUNCTION INTERPOLATES THE SHIPPING HISTOGRAM
C
      DIMENSION A(1)
      ILAST=N
      I=1
      IF(X.GT.A(I).AND.X.LE.A(ILAST)) GO TO 10
      FUNS=0.0
      RETURN
10      I=I+2
      IF(X.LE.A(I).AND.X.GT.A(I-2)) GO TO 20
      GO TO 10
20      FUNS=A(I-1)
      RETURN
      END

```

```

      FUNCTION FUNU (A,X,N)
C
      DIMENSION A(1)
      ILAST=N-1
      I=1
      IF (X.GT.A(I)) GO TO 10
      FUNU=A(2)
      RETURN
10  I=I+2
      IF (X.LE.A(I)) GO TO 20
      IF (I.LT.ILAST) GO TO 10
      GO TO 30
20  IF (A(I).EQ.A(I-2)) GO TO 30
      IF (X.EQ.A(I)) GO TO 30
      FUNU=A(I-1)+(A(I+1)-A(I-1))/(A(I)-A(I-2))*(X-A(I-2))
      RETURN
30  FUNU=A(I+1)
      RETURN
      END

      SUBROUTINE HLOSS (PHIE,R,S,DRDP,GRAD,HKSR)
C
C   THIS SUBROUTINE COMPUTES ABSORPTION(HK), SPREADING(HS),
C   AND REFRACTION(HR) LOSSES.
C
C
C   COMMON /COMX/ CX,ZBM,PHIC,ALPHAC,BION/COMCT/CV,TANPHS
C
C   DATA FMIN/1.0E-6/,FMAX/1.0E12/
      IF (ABS(PHIE).LT.0.0001) GO TO 10
      FOCUS=ABS(CV/CX*R*SIN(PHIE)*DROP)

      GO TO 20
10  FOCUS=ABS(CV/CX*R*TANPHS*CV/GRAD/3000.0)
20  IF (FOCUS.GT.FMIN.AND.FOCUS.LT.FMAX) GO TO 30
      PRINT 40, PHIE,R,DROP,GRAD
      CALL EXIT
30  HKSR=60.0+10.0*ALOG10(FOCUS)+ALPHAC*S

C
      RETURN
C
C   40 FORMAT (5X,'PHIE,R,DRDP,GRAD',4E12.6)
      END

      FUNCTION HSURF (N,NP)
C
C   THIS SUBROUTINE COMPUTES SURFACE REFLECTION LOSS
C
C   N =NUMBER OF REFLECTIONS
C   NP=PATH TYPE
C
C   NP  PATH      REFL'S
C   1    DU        N-1
C   2    DD         N
C   3    UU         N
C   4    UD        N+1
C   5    DP         0
C   6    SR         1

```

```

DATA HS/0.5/
C
GO TO (10,20,20,30,40,50), NP
10 ENS=N-1
GO TO 60
20 ENS=N
GO TO 60
30 ENS=N+1
GO TO 60
40 ENS=0.0
GO TO 60
50 ENS=1.0
C
60 HSURF=ENS*HS
C
RETURN
END
SUBROUTINE PLT (PTEST)
C
C THIS SUBROUTINE PLOTS THE DIRECTIONAL AMBIENT NOISE
C
COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGNO,ZTINC,RT,RTNO,R
2GINC,TARG,TARGO,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMX/CX,ZBM,PH
3IC,ALPHAC,BION
COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
1I,TOTLN,I51,ITST,NTST,THRML,THROB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC

INTEGER PN
C
DATA BERNG/270./,PN/0/
C
CT=CD+80.0
IBR=BERNG
IDGR=DGREH
DBMAX=-1000.
C
IF (HFLAG.LT.0.0 .OR.HFLAG.GT.0.0 .AND.NUMF.GT.1) GO TO 10
READ (5,160) (TYTLE(I),I=1,8)
READ (5,160) (LOCAT(I),I=1,3)
READ (5,160) (DATE(I),I=1,2)
10 CONTINUE
C
C SET UP AND PRINT OUT HEADINGS IF REQUESTED
C
ENCODE (40,160,CAP1) (TYTLE(I), I=1,8)
WSP1=WSPD
FRQ=FREQ*1000.0
ENCODE(64,220,CAP2)(LOCAT(I),I=1,3),FRQ
ENCODE(64,230,CAP3)(DATE(I),I=1,2),ZX
ENCODE(64,240,CAP4) WSP1,TOTLN
ENCODE(64,250,CAP5) ZB1,IDGR,IBR
IF (OUTPT .LE. 0.) GO TO 15
PRINT 200, (CAP2(I),I=1,13)
PRINT 200, (CAP3(I),I=1,13)
PRINT 200, (CAP4(I),I=1,13)

```

```

PRINT 200, (CAPS(I),I=1,15)
PRINT 170
PRINT 180
15 CONTINUE
  IF (PN.LE.0) CALL BALLPT ('PEN 2 BLACKS')
  PN=PN+1
C
C   SKIP SORT IF PROP LOSS READ IN
  IF (PTEST.EQ.1.0) GO TO 151
C
C   SORT DATA AND SET MAX AND MIN GRAPH LIMITS
20 N=0
  J=I51-1
  DO 30 I=1,J,2
    N=N+1
    Q(N)=XANG(I)
    IF (YDB(I).LE.CD) YDB(I)=CD
30  W(N)=YDB(I)
    M=0
    DO 40 I=I51,JP1,2
      M=M+1
      N=JCNT-M+1
      Q(N)=XANG(I)
      IF (YDB(I).LE.CD) YDB(I)=CD
      IF (YDB(I).GE.CT) YDB(I)=CT
40  W(N)=YDB(I)
      N=JCNT
      L=I51+1
      DO 50 I=L,JP1,2
        N=N+1
        Q(N)=XANG(I)
        IF (YDB(I).LE.CD) YDB(I)=CD

        IF (YDB(I).GE.CT) YDB(I)=CT
50  W(N)=YDB(I)
60  M=0
      DO 70 I=2,I51,2
        M=M+1
        N=JP1+1-M
        Q(N)=XANG(I)
        IF (YDB(I).LE.CD) YDB(I)=CD
70  W(N)=YDB(I)
C
C   PRINT ANGLE AND LEVEL WHEN PROP LOSS NOT READ IN IF REQUESTED
  IF (OUTPT .GT. 0.) PRINT 190,Q(1),W(1),Q(JP1),W(JP1)
C
  I=1
80  SUM=0.0
90  I=I+1
    IF (I.GE.JCNT) GO TO 110
    SUM=SUM+ABS(Q(I)-Q(I-1))
    IF (SUM-1.0) 90,100,100
100 L=JP1-I+1
    IF (OUTPT .GT. 0.) PRINT 190,Q(I),W(I),Q(L),W(L)
    GO TO 80
110 IF (OUTPT .GT. 0.) PRINT 210
    CALL BGNPL (1)
    CALL PHYSOR (2.0,1.5)
    CALL TITLE (CAP1,-100,0,0,0,0,7.0,5.5)

```

```

CALL ENDGR (1)
CALL PHYSOR (2.0,1.5)
CALL TITLE (1H,-1,'VERTICAL RECEIVED ANGLE (())DEG())$',100,'NOISE
1 LEVEL (())DB RE (M)PA /STR HZ())$',100,7.0,4.8)
CALL HEIGHT (.1)
CALL ANGLE (90.0)
CALL MESSAG('2$',100,-.7,3.2)
CALL RESET ('ANGLE')
CALL RESET ('HEIGHT')
CALL YTICKS (2)
CALL XTICKS (3)
CALL FRAME
CALL YAXANG (0.0)
CALL INTAXS
CALL MIXALF ('L/CGREEK')
CALL GRAF (-90.0,30.0,90.0,CD,10.0,CT)
CALL RESET ('MIXALF')
CALL HEADIN (CAP2,100,1.4)
CALL HEADIN (CAP3,100,1.4)
CALL HEADIN (CAP4,100,1.4)
CALL HEADIN (CAP5,100,1.4)
CALL CURVE (Q,W,JP1,0)

C
C PLOT TARGET LEVELS IF CALLED FOR AND PROP LOSS NOT READ IN
C
IF (TGDP.LE.0.0.OR.PTEST.EQ.1.0) GO TO 130
KP3=3*KP1
DO 120 I=3,KP3,3
J=I/3
Q(I-2)=TDEG(J)
Q(I-1)=TDEG(J)
Q(I)=TDEG(J)
W(I-2)=CD
W(I-1)=TDB(J)
W(I)=CD

IF (DBMAX.LT.TDB(J)) DBMAX=TDB(J)
IF (W(I-1).LE.CD) W(I-1)=CD
120 IF (W(I-1).GE.CT) W(I-1)=CT
IZTG1=TGDP
ENCODE(64,260,CAP5) IZTG1,RTG
ENCODE(64,270,CAP4) TRECL,TLEV
ENCODE(40,280,CAP6) DBMAX
ENCODE(40,290,CAP7)
CALL HEIGHT (.1)
CALL MESSAG (CAP7,100,4.7,4.45)
CALL HEIGHT (.07)
CALL MESSAG (CAP5,100,4.3,4.25)
CALL MESSAG (CAP4,100,4.4,4.1)
CALL MESSAG (CAP6,100,4.4,3.95)
CALL DASH
CALL CURVE (Q,W,KP3,0)
CALL RESET ('DASH')
CALL RESET ('HEIGHT')
130 CALL ENDPL (0)

C
C RETURN
C
C PRINT OUT ANGLE AND LEVEL WHEN PROP LOSS READ IN IF REQUESTED
C
151 I=0

```



```

152 I=I+1
    J=JP1+1-I
    IF (OUTPT .GT. 0.) PRINT 190, Q(J),W(J),Q(I),W(I)
    IF(J-I-2) 154,153,152
153 IF (OUTPT .GT. 0.) PRINT 190, Q(J-1),W(J-1)
154 CONTINUE
    GO TO 110
C
160 FORMAT (13A5)
170 FORMAT (///12X,9HD/E ANGLE,5X,11HNOISE LEVEL,9X,9HD/E ANGLE,5X11HN
    10ISE LEVEL)
180 FORMAT (14X,5H(DEG),10X,7H(DB) *,12X,5H(DEG),10X,7H(DB) */)
190 FORMAT (3X,F16.1,F14.1,F20.1,F14.1)
200 FORMAT (10X,12A5,A2)
210 FORMAT (/7X,' * DB RE MICROPASCAL**2/STERADIAN HZ'/1H1)
220 FORMAT ('LOCATION',4X,3A5,10X,'FREQ (HZ)',F16.1,'$')
230 FORMAT ('DATE',13X,2A5,10X,'REC DEPTH (FT)',F11.1,'$')
240 FORMAT ('WIND SPEED (KT) ',F9.1,12X,'SECTOR LEVEL (DB)',F8.1,'$')
250 FORMAT ('BOTTOM DEPTH (FT)',F8.1,12X,'HOR SEC (DEG)',I5,' AT',I4,'
    1$')
260 FORMAT ('----',I4,' FT TARGET AT',F7.1,' KYD $')
270 FORMAT (F7.1,' DB REC FROM',F7.1,' DB SOURCES')
280 FORMAT (F7.1,' DB MAX ONE DEGREE AVERAGES')
290 FORMAT (' TARGET DATA $')
    END

```

SUBROUTINE PRINTS

```

C
C THIS SUBROUTINE PRINTS THE INITIAL PARAMETERS
C
C DIMENSION DOUT(24)
C

```

```

COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGNO,ZTINC,RGT,RGTNO,R
2GINC,TARG,TARG0,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4,NUMR,NUMS,NUMBW,NUMP,NUMF
COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JENT,WINDI,TOTL
1I,TOTLN,I51,ITST,NTST,THRML,THROB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC

```

```

C DATA BLANK/5H
C
C SORT SHIPPING HISTOGRAM ARRAY
C
    I=-1
    K=-1
1 K=K+2
    I=I+2
    IF(K.GE.NUMS) GO TO 4
    Q(I)=ASHIP(K)
    Q(I+1)=ASHIP(K+1)
    W(I)=20000.
    W(I+1)=40000.

```

```

2 IF(ASHIP(K+2).GE.0.0) GO TO 1
  IF(ASHIP(K+2).LE.-50.) GO TO 3
  W(I)=-ASHIP(K+2)
  K=K+1
  GO TO 2
3 W(I+1)=-ASHIP(K+2)
  K=K+1
  GO TO 2
4 Q(I)=ASHIP(K)
  W(I)=20000.
  W(I+1)=40000.
  NUMQ=I+1
C
  MAX=MAX0(NUMV,NUMR,NUMH,NUMQ,NUMBW)
C
C   PRINT INPUT VARIABLES
C
  PRINT 70
  PRINT 80, FREQ,ZX,ZTG,PHID,DELPH,BNCS,WSPD,DGREH,BERNG,HOUR,ACTIV,
1RAIN,ALAT,SNPLT,SSPLT,HFLAG,ZTG1,ZTGNO,ZTINC,RGT,RGTNO,RGINC,TARG,
2TARG0,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS
C
C
  PRINT 90
  PRINT 100
C
C   PRINT ARRAYS
C
  DO 60 J=1,MAX,2
  DO 10 I=1,24
10 DOUT(I)=BLANK
  IF (J.GT.NUMV) GO TO 20
  ENCODE(20,110,DOUT(2)) AVEL1(J), AVEL1(J+1)
20 IF (J.GT.NUMH) GO TO 30
  ENCODE(20,110,DOUT(6)) AHB(J),AHB(J+1)
30 IF (J.GT.NUMR) GO TO 40

  ENCODE(20,110,DOUT(10)) ARESP(J),ARESP(J+1)
40 IF (J.GT.NUMQ) GO TO 50
  IF (J.GE.NUMQ-1) GO TO 44
  SUM=W(J)+W(J+1)
  IF (SUM.LT.55000.) GO TO 41
  ENCODE(25,130,DOUT(14)) Q(J),Q(J+1)
  GO TO 50
41 IF (SUM.LT.35000.) GO TO 42
  ENCODE(25,130,DOUT(14)) Q(J),Q(J+1),W(J)
  GO TO 50
42 IF (SUM.LT.15000.) GO TO 43
  ENCODE(25,140,DOUT(14)) Q(J),Q(J+1),W(J+1)
  GO TO 50
43 ENCODE(25,130,DOUT(14)) Q(J),Q(J+1),W(J),W(J+1)
  GO TO 50
44 ENCODE(20,130,DOUT(14)) Q(J)
50 IF (J.GT.NUMBW) GO TO 60
  ENCODE(20,110,DOUT(19)) ABW(J),ABW(J+1)
60 PRINT 120, (DOUT(K),K=1,24)
C
  RETURN

```

```

70 FORMAT (////48X,'** INITIAL PARAMETERS **'//)
80 FORMAT (8X'FREQ =' F6.3' KHZ'5X'ZX   ='F6.1' FT'6X'ZTG   =' F6.1
1' FT'6X'PHID ='F6.1' DEG'5X'DELPH ='F6.1' DEG'/8X'BNCS ='F6.1,9X
2'WSPD ='F6.1' KT '5X'DGREH ='F6.1' DEG'5X'BERNG ='F6.1' DEG'5X'HO
3UR ='F7.2/8X'ACTIV ='F6.1,9X'RAIN ='F6.1' IN/HR'3X'ALAT ='F6.1'
4 DEG'5X'SNPLT ='F6.1,9X'SSPLT ='F6.1/8X'HFLAG ='F6.1,9X'ZTG1 ='F6
5.1' FT'6X'ZTGNO ='F6.1,9X'ZTINC ='F6.1' FT'6X'RG1 ='F6.1' KYD '/'
68X'RG1NO ='F6.1,9X'RG1INC ='F6.1' KYD '4X'TARG ='F6.1,9X'TARG0 ='F
76.1,9X'TARG1 ='F6.1/8X'TARG2 ='F6.1,9X'TARG3 ='F6.1,9X'SHIPD ='F6.
81,9X'SHLFR ='F6.1' NMI'5X'SHLFS ='F6.1//)
90 FORMAT (11X,'VEL PROFILE',9X,'BOTTOM LOSS',10X,'BEAM RESP',9X'SHIP
1PING HISTOGRAM',11X,'BANDWIDTH'//)
100 FORMAT ( 9X,' FT FT/SEC',10X,'DEG DB',12X,'DEG DB',7X,' NM
11 SHIPS KT FT',8X,'HZ DB DOWN'//)
110 FORMAT (F10.1,F8.1)
120 FORMAT (24A5)
130 FORMAT (F7.1,2F6.1,F6.0)
140 FORMAT (F7.1,F6.1,6X,F6.0)
END

```

SUBROUTINE RAPATH (ICZ,PHIS,DELR)

```

C
C THIS SUBROUTINE GENERATES RAY PATHS AND CORRESPONDING
C PROPAGATION LOSSES FOR BB AND CZ MODES. RAY PATH
C COMBINATIONS ARE OBTAINED FOR UP TO NMAX BOTTOM
C REFLECTIONS(OR REFRACTIONS) FOR EACH D/E ANGLE(PHIS).
C
C DIMENSION SX(4), ST(4)
C
C COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGNO,ZTINC,RG1,RG1NO,R
2GINC,TARG,TARG0,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4,NUMR,NUMS,NUMBW,NUMP,NUMF/COMX/CX,ZBM,PHIC,ALPHAC,BION/COMCT/CV,T
5ANPHS

COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
1I,TOTLN,IS1,ITST,NTST,THRML,THRDB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC

C DATA RADCON/57.2957795/
C DATA (SX(I),I=1,4)/-1.0,-1.0,1.0,1.0/
C DATA (ST(I),I=1,4)/-1.0,1.0,-1.0,1.0/
C
C COSPHS=COS(PHIS)
C CV=CX/COSPHS
C CALL VERTEX (ZLO,ZHI)
C IF (ZTG.LT.ZHI.OR.ZTG.GT.ZLO) GO TO 90
C TANPHS=SIN(PHIS)/COSPHS
C PHI=57.296*PHIS
C RAYTRACE FROM SOURCE TO BOTTOM
C CALL RAYTRC (PHIS,ZX,ZBM,SXB,RXB,DRXB,PHIBOT,GRADB,ZEND)
C BRL=FUNU(AHB,RADCON*PHIBOT,NUMH)
C
C RAYTRACE FROM SOURCE UP TO SURFACE
C CALL RAYTRC (-PHIS,0.0,ZX,SXS,RXS,DRXS,PHIE,GRADS,ZEND)

```

```

ZSURF=0.0
NS=0
IF (PHIE.NE.0.0) GO TO 10
IF (ZEND.NE.0.0) NS=1
ZSURF=ZEND
10 CONTINUE
IF (ZTG.EQ.ZX) GO TO 20
C
C RAYTRACE FROM SURFACE (OR VERTEX DEPTH) DOWN TO TARGET
C CALL RAYTRC (PHIE,ZSURF,ZTG,SST,RST,DRST,PHND,GRADT,ZEND)
C
C DIRS IS THE NOISE SOURCE DIRECTIVITY
C
EXPN=5.729*PHND
IF (PHND.LT..3491) EXPN=2.0
DIRS=1.+(SIN(PHND)**EXPN-1.)*(2.*FREQ-.02)
IF (ZEND.EQ.ZTG) GO TO 30
WRITE (6,100) ZEND
GO TO 30
20 SST=SXS
RST=RXS
DRST=DRXS
PHND=PHIS
C
30 CONTINUE
R=2.0*RXB-RXS-RST
C
C LIMIT TO FIVE BOTTOM BOUNCES
C IF (ICZ.EQ.0.AND.NMAX.GT.5) NMAX=5
DO 40 N=1,NMAX
EN=N
HBO=0.0
IF (ICZ.EQ.0) HBO=EN*BRL
RN=2.0*EN*(RXB+RXS)
SN=2.0*EN*(SXB+SXS)
DN=2.0*EN*(DRXB+DRXS)
DO 40 K=1,4
PS=-SX(K)*PHIS
IF (PS.LT.PHMIN.OR.PS.GT.PHMAX) GO TO 40
RK=RN+SX(K)*RXS+ST(K)*RST
C
C MAXIMUM RANGE CHECK
C
SK=SN+SX(K)*SXS+ST(K)*SST
DK=DN+SX(K)*DRXS+ST(K)*DRST
C
C ELIMINATE CAUSTIC BY RESTRICTING DK
C IF (ABS(DK).LE.1.) DK=1.
C
HSB=HSURF(N,K)+HBO
CALL HLOSS (PHND,RK,SK,DK,GRADT,HK)
HK=HK+HSB+FUNU(ARESP,SX(K)*PHI,NUMR)
CALL SUMINT (K,N,NMAX,RK,HK,PHI,DIRS)
40 CONTINUE
C
C DIRECT PATH
C
41 K=5
S=-1.0
IF (ZX-ZTG) 50,80,60

```

```

50 PN=PHIS
   PX=PHIS
   GO TO 70
60 PN=-PHIS
   PX=-PHIS
70 RK=ABS(RXS+S*RST)
   IF(PN.LT.PHMIN.OR.PN.GT.PHMAX) GOTO 80
   SK=ABS(SXS+S*SST)
   DK=ABS(DRXS+S*DRST)
   HS=HSURF(1,K)
   CALL HLOSS (PHND,RK,SK,DK,GRAD,HK)
   HK=HK+HS+FUNK(ARESP,PN,NUMR)
   CALL SUMINT (K,N,NMAX,RK,HK,PHI,DIRS)
80 IF (K.EQ.6) GO TO 90

C
C   SURFACE REFLECTED PATH
   K=6
   N=NMAX+2
   S=1.0
   PN=-PHIS
   PX=PHIS
   GO TO 70

C
90 CONTINUE
   DELR=R-RLAST
   RLAST=R
   RETURN

C
   ENTRY PRERAY
   NTST=0
   ITST=1
   RLAST=200.0
   NMAX=BNCS
   PHMAX=PHID+0.5*DELPH
   PHMIN=PHMAX-DELPH
   IF (PHMIN.GE. 0.0) ITST=2
   PHMAX=PHMAX/RADCON
   PHMIN=PHMIN/RADCON
   RETURN

C
100 FORMAT (//5X,4HZEND,E12.6//)
   END

SUBROUTINE RAYTRC (PHIS,ZUP,ZLO,SP,RP,DRDP,PHIEND,GRAD,ZNP1)

C
C
C   PHIS POSITIVE MEANS DOWN-GOING RAY, TRACES FROM ZUP TO ZLO
C   PHIS NEGATIVE MEANS UP-GOING RAY, TRACES FROM ZLO TO ZUP
C
COMMON /COMA/ AVELP(30),AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(
130),ADANF(30),NUMV,NUMH,NUMR,NUMS,NUMBW,NUMP,NUMF/COMCT/CV,TANPHS

C
C
   RP=0.0
   SP=0.0
   DRDP=0.0
   SINPN=SIN(PHIS)
   COSPN=COS(PHIS)
   PHIN=PHIS
   N=1

```

```

C     TEST FOR NEGATIVE PHISTART
      IF (PHIS.LT.0.0) GO TO 20
C
C     PHIS POSITIVE, DOWN-GOING RAY
      ZN=ZUP
10    NUM=N
      LL=1
      GO TO 40
C
C     PHIS NEGATIVE, UP-GOING RAY
20    ZN=ZLO
30    NUM=NUMV-N
      LL=2
C
40    AZN=AVELP(NUM)
      IF ((ZN.LT.AZN).AND.(LL.EQ.1)) GO TO 50
      IF ((ZN.GT.AZN).AND.(LL.EQ.2)) GO TO 50
      N=N+2
      GO TO (10,30), LL
C
50    ZNP1=AZN
      IF (ABS(ZNP1-ZN).LT.0.001) GO TO 150
      IF (ZNP1.GT.ZLO) ZNP1=ZLO
      IF (ZNP1.LT.ZUP) ZNP1=ZUP
      CN=FUNU(AVELP,ZN,NUMV)
      CNP1=FUNU(AVELP,ZNP1,NUMV)
C
C     TEST FOR VERTEXING RAY
      IF (CNP1.GT.CV) GO TO 60
      PHINP1=ACOS(CNP1/CV)
      IF (PHIS.LT.0.0) PHINP1=-PHINP1
      COSPN1=CNP1/CV
      SINPN1=SQRT(1.0-COSPN1*COSPN1)
      IF (PHIS.LT.0.0) SINPN1=-SINPN1
C
      GO TO 70
C
C     VERTEXING RAY COMPUTATIONS
60    ZNP1=ZN+(CV-CN)/(CNP1-CN)*(ZNP1-ZN)
      CNP1=CV
      PHINP1=0.0
      COSPN1=1.0
      COSPN1=1.0
      SINPN1=0.0
70    DELZ=ZNP1-ZN
C
C     TEST FOR VERTICAL PROFILE
      IF (ABS(CNP1-CN).GT.0.00001) GO TO 80
C
C     VERTICAL PROFILE COMPUTATIONS
      DELRP=DELZ*COSPN/SINPN
      RP=RP+DELRP
      DELSP=DELRP/COSPN
      SP=SP+DELS
      GO TO 90
C
80    GRAD=(CNP1-CN)/DELZ
      RHO=CV/GRAD
      DELRP=RHO*(SINPN-SINPN1)

```

```

      RP=RP+DELRP
      DELSP=RHO*(PHIN-PHINP1)
      IF (DELS.P.LT.0.0) ITEST=1
      SP=SP+DELS
C
90  CONTINUE
      IF (ITEST.NE.0) WRITE (6,160) ZN,ZNP1,CN,CNP1,COSPN,COSPN1,RP,SP
      ITEST=0
C
      TEST FOR LAST TIME THRU FOR VERTEXING RAY
      IF (ABS(CV-CNP1).LT.0.00001) GO TO 120
C
      TEST FOR HORIZONTAL ENTRANCE RAY.
      IF (ABS(PHIN).LT.0.0001) GO TO 100
      DRDP=DRDP-DELRP/SINPN/SINPN1
      GO TO 110
100  DRDP=DRDP-RHO/ABS(SINPN1)
110  CN=CNP1
      ZN=ZNP1
      PHIN=PHINP1
      N=N+2
      SINPN=SINPN1
      COSPN=COSPN1
      IF ((ZNP1.EQ.ZLO).OR.(ZNP1.EQ.ZUP)) GO TO 130
      IF (PHIS) 30,10,10
C
      VERTEXING RAY - MUST EXIT
C
120  CONTINUE
      DRDP=DRDP+RHO/ABS(SINPN)
      PHIEND=0.0
      GO TO 140
130  PHIEND=ABS(PHINP1)
140  RP=RP/3000.0
      SP=SP/3000.0
      DRDP=DRDP/3000.0*TANPHS
C
      RETURN
C
150  WRITE (6,170) ZNP1
C
      RETURN
C
160  FORMAT (1X2HZN10X4HZNP18X2HCN10X4HCNP18X5HCOSPN7X6HCOSPN16X2HRP10X
12HSP/8E12.6)
170  FORMAT (//ZNP1=ZN=E10.4)
      END

      SUBROUTINE READIN (WORD)
C
      THIS SUBROUTINE READS THE INPUT PARAMETERS
C
      INTEGER PAUSE,PARAM,HEADER,ENDATA,PRLOS,WORD,BLANK,ASTRSK,DOT,VNAM
      1E,ANAME,SLSH,DATA,EQU
C
      COMMON /COMV/ VRBL(31)/COMA/ARRAY(30,7),NUMA(7)/COMH/IHDR(16)/COMX
1/CX,ZBM,PHIC,ALPHAC,BION
      COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
1I,TOTLN,I51,ITST,NTST,THRML,THROB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2UB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
30),XP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC

```

```

        DIMENSION VNAME(31), ANAME(7), DATA(5), IDATA(16), JDATA(81)
        DATA (VNAME(I),I=1,31)/5HZX ,5HZTG ,5HRMAX ,5HPHID ,5HDELPH,5HB
1NCS ,5HOUTPT,5HWSPD ,5HDGREH,5HSNPLT,5HSSPLT,5HALAT ,5HBERNG,5HHOU
2R ,5HACTIV,5HRAIN ,5HHFLAG,5HZTG1 ,5HZTGNO,5HZTINC,5HRGT ,5HRGTNO
3,5HRGINC,5HTARG ,5HTARG0,5HTARG1,5HTARG2,5HTARG3,5HSHIPD,5HSHLFR,
45HSHLFS/
        DATA NV/31/
        DATA (ANAME(I),I=1,7)/5HAVELP,5HAHB ,5HARESP,5HASHIP,5HABW ,5HAP
1ROP,5HADANF/
        DATA NA/7/
        DATA PAUSE,PARAM,HEADER,ENDATA,PRLOS ,BLANK,ASTRSK,DOT/'PAUSE','PA
1RAM','HEADE','END=D','PRLOS',' ','*','.'/
        DATA KOMA/' ','/','SLSH/' ','/','EQU/'='/'
        DATA KINN/1/,KOUT/2/
        IF (WORD.NE.PRLOS) WRITE(6,230)
C
C      READ AND PRINT INPUT CARD
10 READ (5,240) KINN,(IDATA(IX),IX=1,16)
    WRITE (6,240) KOUT,(IDATA(IX),IX=1,16)
    K=1
    DECODE(80,250,IDATA) (JDATA(JX),JX=1,80)
C
C      SEARCH FOR BEGINING OF LABEL
20 DO 30 J=K,80
    IF (JDATA(J).EQ.KOMA) JDATA(J)=BLANK
    IF (JDATA(J).EQ.SLSH) JDATA(J)=BLANK
    IF (JDATA(J).NE.BLANK) GO TO 40
30 CONTINUE
    GO TO 10
C
C      FIND THE END OF THE LABEL
40 DO 50 K=J,80

    IF (JDATA(K).EQ.EQU) GO TO 60
    IF (JDATA(K).EQ.BLANK) GO TO 60
    IF (JDATA(K).EQ.KOMA) GO TO 60
    IF (JDATA(K).EQ.SLSH) GO TO 60
50 CONTINUE
    GO TO 80
C
C      PACK UP TO FIVE CHARACTERS OF LABEL INTO CONTROL WORD
60 K5=K
    JDATA(K)=BLANK
    IF (K-J.GT.5) K5=J+4
    DATA (1)=BLANK
    ENCODE(10,250,DATA) (JDATA(JX),JX=J,K5)
    WORD=DATA(1)
    IF (WORD.EQ.PAUSE.OR.WORD.EQ.ENDATA.OR.WORD.EQ.PRLOS) RETURN
    IF (WORD.EQ.HEADER) GO TO 210
    IF (WORD.EQ.PARAM) GO TO 20
    IF (WORD.EQ.ASTRSK) GO TO 200
C
C      SEE IF LABEL IS A PARAM OR ARRAY
DO 70 I=1,NV
    IF (WORD.EQ.VNAME(I)) GO TO 90
    IF (I.GT.NA) GO TO 70
    IF (WORD.EQ.ANAME(I)) GO TO 180
70 CONTINUE

```



```

C      MUST BE AN ERROR
80  WRITE (6,260) WORD
    GO TO 220

C
C      SET UP TO PROCESS SINGLE VALUED PARAMETER
90  ASSIGN 170 TO JAD
    KIND=1

C
C      PUT VALUE STRING TOGETHER
100 DO 110 J=K,80
    IF (JDATA(J).EQ.BLANK) GO TO 120
110 CONTINUE
120 DO 130 K=J,80
    IF (JDATA(K).EQ.KOMA) GO TO 20
    IF (JDATA(K).EQ.SLSH) GO TO 20
    IF (JDATA(K).EQ.EQU) JDATA(K)=BLANK
    IF (JDATA(K).NE.BLANK) GO TO 140
130 CONTINUE
    VAL=0.0
    GO TO 10
140 NDOT=0
    DO 150 J=K,80
    IF (JDATA(J).EQ.BLANK) GO TO 160
    IF (JDATA(J).EQ.DOT) NDOT=NDOT+1
    IF (JDATA(J).EQ.KOMA) GO TO 160
    IF (JDATA(J).EQ.SLSH) GO TO 160
150 CONTINUE
    J=81
160 KAR=JDATA(J)
    JDATA(J)=BLANK

C
C      INSERT A DECIMAL POINT IF NUMBER HAD NONE
IF (NDOT.EQ.0) JDATA(J)=DOT
DATA (1)=BLANK
DATA (2)=BLANK

    ENCODE(10,250,DATA)(JDATA(JX),JX=K,J)
    JDATA(J)=KAR

C
C      FORM THE VALUE AS A REAL
DECODE (10,270,DATA) VAL
K=J

C
C      GO TO STORE PARAMETER OR ARRAY ELEMENT
GO TO JAD, (170,190)
C      STOR SINGLE VALUE IN THE RIGHT SPOT AND GO BACK FOR MORE IF CARD N
C
170 VRBL(I)=VAL
    IF (K.GE.80) GO TO 10
    GO TO 20

C
C      SET UP TO PROCESS ARRAY ELEMENTS
180 ASSIGN 190 TO JAD

C
C      VFLAG IS THE VELOCITY PROFILE RERUN FLAG FOR CURVATURE
C
IF (I.EQ.1) VFLAG=0.0
KIND=2
NX=0
GO TO 100

```

C UPDATE NUMBER OF ELEMENTS FOR THIS ARRAY AND STORE THE VALUE

```
190 NX=NX+1
   ARRAY(NX,I)=VAL
   NUMA(I)=NX
   IF (K.GE.80) GO TO 10
   IF (JDATA(K).EQ.BLANK) GO TO 100
   GO TO 20
200 IF (KIND.EQ.2) GO TO 100
   GO TO 80
```

C

C READ AND STORE THE TITLE INTO THE COMMON AREA

```
210 READ (5,240) KINN,(IHDR(IX),IX=1,16)
   WRITE (6,240) KOUT,(IHDR(IX),IX=1,16)
   GO TO 10
```

C

220 CALL EXIT

C

```
230 FORMAT (1H1//)
240 FORMAT (T*,16A5)
250 FORMAT (80A1)
260 FORMAT (//10X'ERROR IN INPUTS - ',A5)
270 FORMAT (F10.0)
   END
```

 SUBROUTINE SCLNCE (TMIN,TMAX,ND,SNMIN,SNMAX,SNINC)

C

C THIS SUBROUTINE PRODUCES A NICE SET OF SCALE NUMBERS FOR A PLOT AX
C INPUTS ARE THE DATA MIN 'TMIN', DATA MAX 'TMAX', AND THE REQUIRED N
C INTERVALS ALONG THE AXIS 'ND'. THE ROUTINE OUTPUTS A NEW MIN 'SNMI
C MAX 'SNMAX', AND INCREMENT 'SNINC' FOR ND+1 READABLE VALUES

C

 DIMENSION POWTEN(7), TICVAL(8)

```
      DATA (POWTEN(I),I=1,7)/10.,100.,1000.,10000.,100000.,1000000.,1000
10000./
```

C

C ALLOWABLE VALUES ARE SOME 'TICVAL' TIMES SOME POWER OF TEN
C DATA (TICVAL(I),I=1,8)/0.5,1.0,1.5,2.0,5.0,10.,15.,20./
C DIV=ND

C

C SET UP ROUNDING VALUE
C ROUND=0.01
C ST=TMIN
C IF (ST.LT.0.0) ROUND=ROUND-1.0

C

C DETERMINE FIRST GUESS AT INCREMENT
C D=(TMAX-TMIN)/DIV
C P=1.0
C IF (D.LT.10.0) GO TO 30

C

C REDUCE INCREMENT TO RANGE (1 - 10)
C DO 10 I=2,7
C IF (POWTEN(I).GT.D) GO TO 20
10 CONTINUE
20 P=POWTEN(I-1)
30 D=D/P-0.01

```

C      FIND FIRST ALLOWABLE TICVAL
      DO 40 I=1,7
      IS=I
      IF (TICVAL(I).GE.D) GO TO 50
40    CONTINUE
C
C      COMPUTE NEW INCREMENT
50    D=TICVAL(IS)*P
      ST=D*AIN(TST/D+ROUND)
C
C      TEST NEW INCREMENT TO MAKE SURE DATA WILL FIT IN NEW RANGE
      TEST=ST+(DIV+0.01)*D
      IF (TEST.GE.TMAX) GO TO 60
      IS=IS+1
      GO TO 50
C
C      COMPUTE NEW MIN AND ADJUST
60    ST=ST-AINT((DIV+(ST-TMAX)/D)/2.0)*D
      IF (ST.TMIN) 70,70,80
70    ST=0.0
80    SNMIN=ST
C
C      COMPUTE NEW MAX
      SNMAX=ST+D*DIV
      SNINC=D
      RETURN
      END

      SUBROUTINE SHIPIN (SHIPI)
C
C      THIS SUBROUTINE CONSTRUCTS THE SHIPPING DENSITY NOISE INTENSITY
C      HISTOGRAM
C
      COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCs,OUTPT,WSPD,DGREH,SNPLT,S
1SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGNO,ZTINC,RGT,RGTNO,R
2GINC,TARG,TARGO,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4,NUMR,NUMS,NUMBW,NUMP,NUMF
      COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
1I,TOTLN,IS1,ITST,NTST,THRML,THRDB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC
C
      REAL LMULT
C
C
C      J=0
      K=NUMS-2
C
      DO 40 I=1,K,2
C
C      INITIALIZE PARAMETERS
C
      LMULT=1.0
      SMULT=1.0

```

```

C      STORE SHIPS AND FIRST RANGE OF RANGE INCREMENT
C
      RANG1=ASHIP(I)
      SHIPS=ASHIP(I+1)
      J=J+2
10    IF(ASHIP(I+2).GE.0.0) GO TO 30
C
C      CALCULATE INTENSITY MULTIPLICATION FACTORS FOR SHIP SPEED
      AND LENGTH
      IF(ASHIP(I+2).LE.-50.) GO TO 20
      ADSHP=50.*ALOG10(-ASHIP(I+2)/12.)
      SMULT=2.**(ADSHP/3.0103)
      I=I+1
      GO TO 10
20    ADLNG=20.*ALOG10(-ASHIP(I+2)/300.)
      LMULT=2.**(ADLNG/3.0103)
      I=I+1
      GO TO 10
C
C      CALCULATE AREA AND CONSTRUCT SHIPPING DENSITY NOISE INTENSITY
      HISTOGRAM
30    RANG2=ASHIP(I+2)
      AREA=.0087266*(RANG2**2-RANG1**2)*DGREH
      DENS=SHIPS*10000./AREA
      ASHI1(J-1)=RANG1*2.
      ASHI1(J)=DENS*SHIPS*SMULT*LMULT
40    CONTINUE
C
      NMS2=J+1
      ASHI1(J+1)=RANG2*2.
C
      RETURN
C
      END

```

SUBROUTINE SSPLIT

```

C
C      THIS SUBROUTINE WILL PRODUCE A CALCOMP PLOT OF A SOUND SPEED PROFI
C      'DISSPLA'. THE DEPTH VALUES ARE ALONG THE X AXIS AND THE VEL. VALUE
C      THE Y AXIS. THE PLOT WILL BE DRAWN ON A 8.5 X 11 INCH PAGE WITH AU
C
      DIMENSION LABEL(14), X(15), Y(15)
      COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1      ISPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGNO,ZTINC,RGT,RGTNO,R
2      GINC,TARG,TARGO,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3      ,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4      ,NUMR,NUMS,NUMBW,NUMP,NUMF
      COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
1      I,TOTLN,I51,ITST,NTST,THRML,THRDB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2      DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
3      30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4      ,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC
      DATA NP/100/
C
      NP=NP+1
      NV=NUMV/2

```

```

C      BEGIN PLOT
      CALL BGNPL (NP)
      IF (HFLAG.LT.0.0.OR.SNPLT.GT.0.0) GO TO 10
      READ (5,50) (LOCAT(I),I=1,3)
      READ (5,50) (DATE(I),I=1,2)
10     CONTINUE
      ENCODE(40, 70,CAP2)(LOCAT(I),I=1,3)
      ENCODE(40, 80,CAP3)(DATE(I),I=1,2)
      CALL TITLE (1H ,-1,0,0,0,0,5.0,3.0)

C
C      FIND RANGE OF SOUND SPEED
      YMIN=6000.0
      YMAX=0.0
      DO 20 I=1,NV
      X(I)=AVAL1(2*I-1)
      Y(I)=AVAL1(2*I)
      IF (Y(I).GT.YMAX) YMAX=Y(I)
      IF (Y(I).LT.YMIN) YMIN=Y(I)
20     CONTINUE

C
C      FIGURE OUT NICE SCALES FOR X AND Y
      CALL SCLNCE (YMIN,YMAX,5,YMN,YMX,YINC)
      CALL SCLNCE (X(1),X(NV),10,XMIN,XMAX,XINC)

C
C      SET UP AND DRAW THE X AND Y AXIES
      CALL BANGLE (180.0)
      CALL BSHIFT (4.5,3.0)
      CALL GRAF (XMIN,XINC,XMAX,YMN,YINC,YMX)

C
C      SET UP THE REQUIRED ALPHABET
      CALL BASALF ('L/CSTD')
      CALL MIXALF ('STANDARD')
      CALL HEIGHT (0.1)

C
      CALL CURVE (X,Y,NV,0)

C
C      MAKE AND LABEL X AXIS

      XTMP=XMIN
      DO 30 I=1,11
      IXTM=XTMP
      ENCODE (10,60,LABEL(13)) IXTM
      LABEL(I)=LABEL(14)
30     XTMP=XTMP+XINC
      CALL XAXANG (90.)
      CALL XLBAXS (LABEL,1,11,5.0,' $',100,0.0,0.0)

C
C      MAKE AND LABEL Y AXIS
      YTMP=YMN
      DO 40 I=1,6
      IYTM=YTMP
      ENCODE (10,60,LABEL(13)) IYTM
      LABEL(I)=LABEL(14)
40     YTMP=YTMP+YINC
      CALL YLBAXS (LABEL,1,6,3.0,'(SOUND SPEED) - FT/SEC$',100,0.0,0.0)
      CALL ANGLE (-180.)
      CALL MESSAG ('(DEPTH) - FEET',14,3.0,-0.6)
      CALL ANGLE (90.)
      CALL RESET ('BASALF')
      CALL MESSAG (CAP2,100,-1.25,.45)
      CALL MESSAG (CAP3,100,-1.0,.45)

```

```

C
C   FRAME AND END PLOT
C   CALL FRAME
C   CALL ENDPL (0)
C   RETURN

C
50 FORMAT (3A5)
60 FORMAT (I10)
70 FORMAT ('LOCATION',4X,3A5,'$')
80 FORMAT ('DATE',13X,2A5,'$')
END

SUBROUTINE SUMINT (K,N,NMAX,RK,HK,PHI,DIRS)

C
C   THIS SUBROUTINE COMPUTES THE NOISE INTENSITY ALONG EACH RAY
C
COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGNO,ZTINC,RGT,RGTNO,R
2GINC,TARG,TARG0,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4,NUMR,NUMS,NUMBW,NUMP,NUMF/COMX/CX,ZBM,PHIC,ALPHAC,BION
COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
1I,TOTLN,I51,ITST,NTST,THRML,THRDB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC

C
DATA RADCON/57.2957795/

C
C   TEST FOR FIRST BOUNCE
C
IF (NTST.EQ.1) GO TO 10
NTST=1
DOWNI=0.0

UPI=0.0
10 ABSFI=ABS(PHINC)
RG=RK

C
SHIP1=FUNS(ASHI1,RG,NMS2)
DR=ABS(RNG(K,N)-RK)
IF (RK.EQ.0.0.OR.RNG(K,N).EQ.0.0) DR=5.0
HL=.5*(HK+HLOS(K,N))
HLOS(K,N)=HK
RNG(K,N)=RK

C
C   TEST FOR FIRST TIME THRU
C
IF (ITST.EQ.1.AND.K.EQ.6) GO TO 70
IF (ITST.EQ. 2 .AND. K .EQ. 2 .AND. N .EQ. NMAX) GO TO 70
IF (ITST.GE.1) GO TO 50
AREA=DGREH*.00873*DR*(2.0*RG+DR)
XZZ=AREA*0.5** (HL/3.0103)

C
C   ASSUME WIND NOISE DIPOLE DIRECTIONALITY PATTERN AT HIGH FREQ

```

```

C      WRECI=XZZ*WINDI*DIRS
      SRECI=XZZ*SHIP1
      TOTI=WRECI+SRECI
      IF (K.GE. 3) GO TO 20
      DOWNI=DOWNI+TOTI
      GO TO 30
20    UPI=UPI+TOTI
      PHUP=-PHI
30    CONTINUE
      IF (K.EQ. 2 .AND. ITST.EQ. -1 .AND. N.EQ. NMAX) GO TO 31
      IF (K.NE.6) GO TO 50
31    CONTINUE
      NTST=0
      DOWNI=DOWNI+(THRML+BION)*ABSFI
      UPI=UPI+(THRML+BION)*ABSFI
      TOTLI=TOTLI+DOWNI+UPI
      DBDN=10.0*ALOG10(DOWNI)+184.8856
      DBUP=10.0*ALOG10(UPI)+184.8856

C      NORMALIZE LEVEL TO ONE STERAD AND STORE ARRAYS FOR NOISE PLOT
C
C      JCNT=JCNT+1
C
C      CALCULATE PER STERADIAN CORRECTION FACTOR
C      ADDB=-10.0*ALOG10(ABSFI*COS(PHI/RADCON)*.109662)
      J=2*JCNT-1
      JP1=J+1
      XANG(J)=PHUP
      YDB(J)=DBUP+ADDB
      XANG(JP1)=PHI
      YDB(JP1)=DBDN+ADDB
40    CONTINUE
      IF (SNPLT.GT.0.0 .OR. OUTPT.LE. 0.0) GO TO 50
      PRINT 90, PHI,PHINC,RNG(4,NMAX),HLOS(4,NMAX),DBUP,DBDN,ADDB
50    RETURN

C
70    ITST=0
      IF (K.EQ. 2) ITST=-1
      RETURN

C
90    FORMAT (16X,8F12.3)
      END

      SUBROUTINE SURFI

C      THIS SUBROUTINE CALCULATES THE INTENSITIES OF ALL THE SURFACE
C      NOISE GENERATORS
C
      COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGNO,ZTINC,RGT,RGTNO,R
2GINC,TARG,TARGO,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4,NUMR,NUMS,NUMBW,NUMP,NUMF/COMX/CX,ZBM,PHIC,ALPHAC,BION
      COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
1I,TOTLN,I51,ITST,NTST,THRML,THRDB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC

```

```

C
C WINT, RAINI AND ASHIP(2N) ARE SOUND INTENSITY DENSITIES DUE
C TO WIND, RAIN AND SHIPS. (WATTS/M**2)/KYDS**2
C
DO 10 J=1,11
BB1=ALOG10(FRE3(J))
10 DB(J)=BB1*(4.22*BB1-33.4)-10.9
CALL BWIDTH (THRML)
C
C ESTABLISH LOWEST THERMAL LEVEL/VERT DEG/HOR DEG
C
THRML=THRML*DGREH*1.5432E-5
THRDB=10.0*ALOG10(THRML+BION)+84.8856
C
C ESTABLISH MINIMUM SCALE CD FOR NOISE PLOT
C AND CONVERT TO DB/MICROPASCALS
C
CD=20.
IF (THRDB.LE.-80.0) CD=10.
IF (THRDB.LE.-90.0) CD=0.0
IF (THRDB.LE.-100.0) CD=-10.0
IF (THRDB.LE.-110.0) CD=-20.0
C
WINT=0.0
IF (WSPD.EQ.0.0) GO TO 30
C
C WMULT IS WIND SPEED DEPENDENCE
C
WMULT=.065*WSPD**.7
DO 20 J=1,11
BB1=ALOG10(FRE3(J))
C CORW IS THE SLOPE AND LEVEL CORRECTION FOR SURFACE NOISE SPECTRA
C EXCEPT SHIPPING NOISE
C
CORW=29.6*10.**(1.43E-5*FRE3(J)**1.54)
WLEV1=-18.*BB1-4.
WLEV2=9.66*BB1*BB1-43.99*BB1+23.33
20 DB(J)=WLEV1+(WLEV2-WLEV1)*WMULT+CORW
C
CALL BWIDTH (WINT)
30 RAINI=0.0
IF (RAIN.EQ.0.0) GO TO 50
C
C CALCULATE NOISE DUE TO RAIN
C
ARAIN=ALOG10(RAIN)
DO 40 J=1,11
BB1=ALOG10(FRE3(J))
CORW=.0667*BB1**4.106+29.64
40 DB(J)=CORW+5.5*BB1+14.5*ARAIN-49.5
CALL BWIDTH (RAINI)
50 WINDI=WINT+RAINI
C
C CORL IS THE MODEL DISTORTION FACTOR FOR OMNI SOURCES
C
IF (SHIPD.EQ.0.0) GO TO 90
DO 60 J=1,11
BB1=ALOG10(FRE3(J)-2.*SHIPD+12.)
CORL=14.*FREQ+8.74+4.*SHIPD
SL1=-1.0*BB1+1.16

```



```

        SL2=3.3*BB1-6.274
        DB(J)=-15.-10.*ALOG10(10.**SL1+10.**SL2)+CORL
        IF(SHLFS .LE. 0.0) DB(J)=DB(J)+5.
60 CONTINUE
C
        CALL BWIDTH (SHIPI)
70 CONTINUE
C
C      CONSTRUCT SHIPPING DENSITY NOISE INTENSITY HISTOGRAM
C
        CALL SHIPIN (SHIPI)
C
        RETURN
90 SHIPI=0.0
   GO TO 70
   END

SUBROUTINE TARGET
C
C      THIS SUBROUTINE COMPUTES THE DIRECTIONAL SIGNAL
C      LEVEL RECEIVED FROM A TARGET
C
C
      COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGN0,ZTINC,RGT,RGTNO,R
2GINC,TARG,TARG0,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4,NUMR,NUMS,NUMBW,NUMP,NUMF/COMX/CX,ZBM,PHIC,ALPHAC,BION/COMCT/CV,T
5ANPHS
      COMMON HLOS(6,60),RNG(6,60),XANG(300),YDB(300),JP1,JCNT,WINDI,TOTL
11,TOTLN,I51,ITST,NTST,THRML,THRDB,CD,ZB1,VFLAG,BANDL(11),FRE3(11),
2DB(11),BANDW,LOCAT(3),DATE(2),AVEL1(30),ASHI1(30),TDB(180),TDEG(18
30),KP1,TRECL,TLEV,TGDP,RTG,Q(300),W(300),TYTLE(8),CAP1(8),CAP2(13)
4,CAP3(13),CAP4(13),CAP5(13),CAP6(8),CAP7(8),NMS2,FREQ,PHINC
C
      DATA RADCON/57.29577/

C
C      CX=FUNU(AVELP,ZX,NUMV)
C
C      CTG=FUNU(AVELP,TGDP,NUMV)
      PHITG=0.0001
      IF (CX.LE.CTG) PHITG=ACOS(CX/CTG)
      TDEG(1)=90.
      TDB(1)=-300.
      KP1=1
      UP=1.0
      RUN=0.0
      YLAST=0.0
      TRECI=1.0E-30
      PRINT 230
      IF (TARG.LT.0.0) GO TO 20
C
C      CALCULATE BROADBAND TARGET ENERGY
C
      DO 10 J=1,11
      BB1=ALOG10(FRE3(J))
10 DB(J)=TARG0+TARG1*BB1+TARG2*BB1**2+TARG3*BB1**3

```

```

      CALL BWIDTH (TRGTI)
      GO TO 30
20  TRGTI=3.2467E-9*10.** (TARGO/10.)
30  CONTINUE
C
C      TLEV IS EFFECTIVE TARGET LEVEL IN DB
C
      TLEV=10.*ALOG10 (TRGTI)+84.8856
C
C      FIND BOTTOM GRAZING ANGLE
C      PHIBM=ACOS (CX/AVELP (NUMV))
C
40  PHI1=DELPH/(2.0*RADCON)
      IF (RGT.GE.500.) GO TO 200
      CZ=-1.0
      ITEST=0
      DPHI=-.05
      TINT1=1.0E-35
50  IR=0
      XTRAB=0.0
C
C      TEST FOR RAY WITHIN ACCEPTABLE ANGULAR LIMITS
C
      IF (PHI1.GT.PHIBM.AND.CZ.GT.0.0) GO TO 180
      IF (PHI1.LT.0.0.OR.PHI1.LT.PHIBM.AND.CZ.LT.0.0) GO TO 200
      COSP1=COS (PHI1)
      CV=CX/COSP1
      CALL VERTEX (ZLO,ZHI)
C
C      TEST FOR RAY UNABLE TO REACH TARGET DEPTH
C
      IF (ZLO.LT.(TGDP+5.).OR.ZHI.GT.(TGDP-5.)) GO TO 160
      TANPHS=TAN (PHI1)
      CALL RAYTRC (-PHI1,0.0,ZX,S1,R1,DRDP1,PHND1,GRAD1,ZEND1)
      CALL RAYTRC (PHND1,ZEND1,TGDP,S2,R2,DRDP2,PHND2,GRAD2,ZEND2)
      CALL RAYTRC (PHI1,ZX,ZBM,S3,R3,DRDP3,PHND3,GRAD3,ZEND3)
C
C      TEST FOR RELATIVE POSITION OF TARGET
C
      IF (TGDP-ZX) 60,60,70

C      TARGET ABOVE RECEIVER
60  RG1=2.*R2
      RG2=2.*(R3+R1-R2)
      RSUP=R1-R2
      RSDN=R1-R2+2.*R3
      E1=1.0
      E2=0.0
      GO TO 80
C      TARGET BELOW RECEIVER
70  RG1=2.*(R3+R1-R2)
      RG2=2.*R2
      RSUP=R1+R2
      RSDN=R2-R1
      E1=0.0
      E2=1.0
C
C      TEST FOR UP-GOING OR DOWN-GOING RAY
C

```

```

      80 IF (UP) 90,90,100
C     DOWN GOING RAY
      90 RSTRT=RSDN
        F1=1.0
        F2=0.0
        GO TO 110
C     UP GOING RAY
      100 RSTRT=RSUP
        F1=0.0
        F2=1.0
      110 SIZE=RG1+RG2
C
C     CALCULATE NUMBER OF CYCLES TO TARGET
C
      IF (ITEST.EQ.0) N=(RTG-RSTRT)/SIZE
      Y0=N
      RNODE=Y0*SIZE+RSTRT
      ITEST=1
C
C     TEST TO GET THE SAME NODE AS FIRST TIME THRU AND LOCATION OF
C     TARGET IN CYCLE
C
      IF (IR) 130,120,140
      120 IF (RNODE+RG1-RTG) 130,130,140
      130 RNOD1=RNODE+RG1
        G1=1.0
        IR=-1
        GO TO 150
      140 RNOD1=RNODE
        G1=0.0
        IR=1
      150 CONTINUE
C
C     TEST FOR RAY WITHIN RANGE INCREMENT
C
      DR2=RNOD1-RTG
      IF (DR2) 160,170,170
      160 PHI1=PHI1+DPHI*(1.0+PHI1*.13*(1.+CZ))
      GO TO 50
C
C     CALCULATE LOSSES
C
      170 CONTINUE
      ITEST=0
      SNOD1=2.*Y0*(S1+S3)+E1*(S1-S2+F1*2.*S3)+E2*(S2+S1*(F2-F1))+2.*G1*(
1E1*S2+E2*(S3+S1-S2))
      FNOD1=2.*Y0*(DRDP1+DRDP3)+E1*(DRDP1-DRDP2+F1*2.*DRDP3)+E2*(DRDP2+D
1RDP1*(F2-F1))+2.*G1*(E1*DRDP2+E2*(DRDP3+DRDP1-DRDP2))
      CALL HLOSS (PHND2,RNOD1,SNOD1,FNOD1,GRAD2,HLOS1)
C
C     TEST FOR BOTTOM BOUNCE
C
      BLOS=0.0
      SLOS=0.0
      IF (ZEND3.GE.ZBM) BLOS=FUNU(AHB,PHND3,RADCON,NUMH)
      IF (ZEND1.LE.0.0) SLOS=Y0*.5
      IF (UP.LE.0.0.AND.TGDP.LE.ZX.OR.TGDP.GE.ZX.AND.G1.EQ.1.0) XTRAB=1.
      10 PHI=-UP*PHI1*RADCON
      TLOS=(Y0+XTRAB)*BLOS+HLOS1+FUNU(ARESP,PHI,NUMR)

```

```

C
C      SUM INTENSITY OVER ONE DEGREE AND STORE
C
      KP1=KP1+1
      TDB(KP1)=TLEV-TLOS=SLOS
      TDEG(KP1)=PHI
      TGINT=10.**(TDB(KP1)/10.)
      TINT1=TINT1+TGINT
      TRECI=TRECI+TGINT
      IF (ABS(TDEG(KP1)-TDEG(KP1-1)).LT.1.0) GO TO 210
      IF (Y0.NE.YLAST.AND.Y0.GT.5.) TINT1=2.*TINT1
      TDB(KP1)=10.*ALOG10(TINT1)
      PRINT 240, TDEG(KP1),TDB(KP1)
      TINT1=1.0E-35
      IF (RNOD1.EQ.RSTRT) GO TO 180

C
C      TEST FOR CHANGE FROM UP TO DOWN GOING RAYS
C
      IF (PHI1.GT.PHIBM.AND.CZ.LT.0.0.OR.PHI1.LT.PHIBM.AND.CZ.GT.0.0) GO
1    TO 110
      IF (CZ.GT.0.0) GO TO 180
      GO TO 200
180  RUN=RUN+1.0
      UP=-1.0
      ITEST=0
      IF (RUN-1.0) 40,40,190
190  TRECL=10.*ALOG10(TRECI)
      PRINT 220, TRECL,ZX,TLEV,FREQ,TGDP,RTG
      RETURN

C
C      START BOUNCE FREE RAYS
C
200  PHI1=PHI1G
      CZ=1.0
      DPHI=2.5/RTG
      IF (DPHI.GT..005) DPHI=.005
      ITEST=0
      TINT1=1.0E-35
      GO TO 50
210  KP1=KP1-1
      YLAST=Y0
      GO TO 110

C
220  FORMAT (//F12.1, ' DB SIG REC AT',F7.0, ' FT FROM' ,F7.1, ' DB TARGET
1    AT',F6.4, ' KHZ',F6.0, ' FT DEPTH',F8.3, ' KYD' /1H1)
230  FORMAT (/30X, 'REC ANGLE',5X, 'TARGET LEVEL'/33X, 'DEG',10X, 'DB/M PA
1    /)
240  FORMAT (28X,F10.3,5X,F10.2)
      END

      SUBROUTINE VERTEX (ZLO,ZHI)

C
C      THIS SUBROUTINE COMPUTES HIGHEST AND LOWEST DEPTHS REACHED BY RAY
C
      COMMON /COMV/ ZX,ZTG,RMAX,PHID,DELPH,BNCS,OUTPT,WSPD,DGREH,SNPLT,S
1    SPLT,ALAT,BERNG,HOUR,ACTIV,RAIN,HFLAG,ZTG1,ZTGNO,ZTINC,RGT,RGTNO,R
2    GINC,TARG,TARGO,TARG1,TARG2,TARG3,SHIPD,SHLFR,SHLFS/COMA/AVELP(30)
3    ,AHB(30),ARESP(30),ASHIP(30),ABW(30),APROP(30),ADANF(30),NUMV,NUMH
4    ,NUMR,NUMS,NUMBW,NUMP,NUMF/COMCT/CV,TANPHS

```

```

C
  N=-1
10 N=N+2
   ZN=AVELP(N)
   IF (ZN.LT.ZX) GO TO 10
   NST=N-2
C
C   SEARCH UP
   ZNP1=AVELP(N)
   CNP1=AVELP(N+1)
   N=NST
20 ZN=AVELP(N)
   CN=AVELP(N+1)
   IF (CN.GE.CV) GO TO 30
   CNP1=CN
   ZNP1=ZN
   N=N-2
   IF (N.GT.0) GO TO 20
   ZHI=0.0
   GO TO 50
30 GRAD=(CN-CNP1)/(ZN-ZNP1)
   IF (GRAD.EQ.0.0) GO TO 40
   ZHI=ZNP1+(CV-CNP1)/GRAD
   GO TO 50
40 ZHI=ZNP1
C
C   SEARCH DOWN
50 CONTINUE
   ZN=AVELP(NST)
   CN=AVELP(NST+1)
   N=NST+2
60 ZNP1=AVELP(N)
   CNP1=AVELP(N+1)
   IF (CNP1.GE.CV) GO TO 70
   CN=CNP1
   ZN=ZNP1
   N=N+2
   IF (N.LT.NUMV) GO TO 60
   ZLO=AVELP(N-2)
   RETURN
70 GRAD=(CNP1-CN)/(ZNP1-ZN)
   IF (GRAD.EQ.0.0) GO TO 80

   ZLO=ZN+(CV-CN)/GRAD
   RETURN
80 ZLO=ZN
   RETURN
C
  END

```

